

AD-A112 680

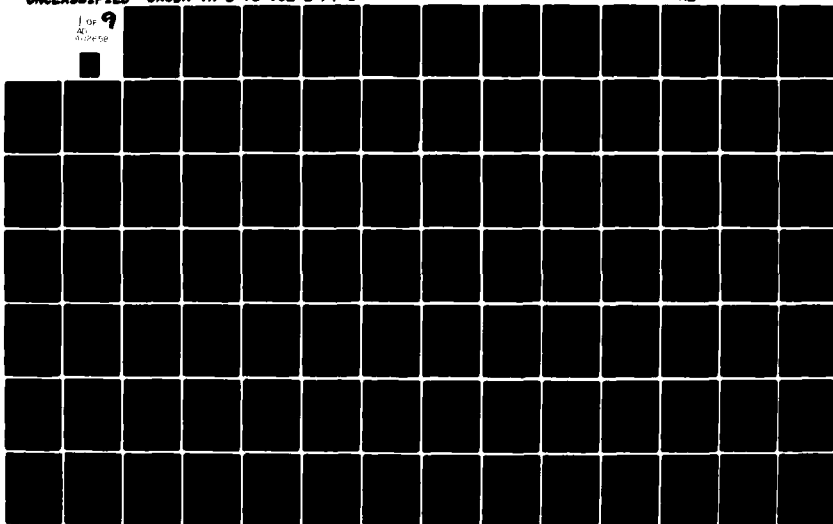
ARMY COMBINED ARMS COMBAT DEVELOPMENTS ACTIVITY FORT--ETC F/8 15/7  
DIVVAB MODEL DOCUMENTATION. VOLUME II. PROGRAMMER/ANALYST MANUAL--ETC(U)  
JAL 76

UNCLASSIFIED

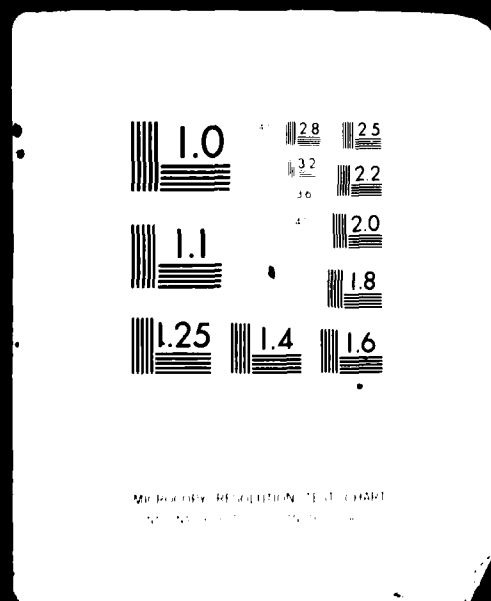
CACDA-TR-8-76-VOL-2-PY-1

NL

For  
AC  
Address



1 OF 9  
AD  
A112658



②

Technical Report TR 8-76  
July 1976

Directorate of Combat Operations Analysis  
US Army Combined Arms Combat Developments Activity  
Fort Leavenworth, Kansas 66027

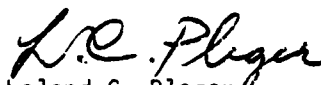
DIVWAG MODEL DOCUMENTATION  
VOLUME II  
PROGRAMMER/ANALYST MANUAL

ACN 21704

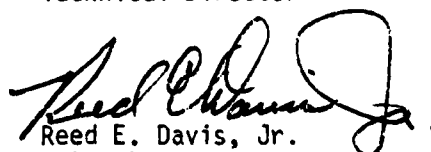
Approved by:



Thomas E. deShazo, Jr.  
Colonel, FA  
Chief, War Games Division



Leland C. Pleger  
Technical Director



Reed E. Davis, Jr.  
Colonel, IN  
Director

This document has been approved  
for public release and its  
distribution is unlimited.

DTIC  
ELECTRONIC  
MAR 17 1982  
E

8 2 17 126

AL 4112658

DTIC FILE COPY

0-0509

## ABSTRACT

This documentation provides a complete description of the Division War Game (DIVWAG) Model as it exists on 1 April 1976. The documentation is composed of an Executive Summary (Volume I), an Analyst/Programmer Manual (Volume II), and the Planner/User Manual (Volume III). Described within the volumes are the model design and development; application; capabilities; limitations; facility, equipment, and personnel requirements; data input requirements; mathematical and logical processes; program descriptions; output descriptions; user instructions; and diagnostic messages. This documentation was originally produced in April 1973 by Computer Sciences Corporation (CSC) under Contract DAAG 11-70-0875.

Acco  
NFI  
NFI  
L  
L  
*Ltr on file per*  
L  
L  
L  
Dirt  
A



NOTES

## CONTENTS

### PART 1

	<u>Page</u>
TITLE PAGE . . . . .	i
ABSTRACT . . . . .	iii
CONTENTS . . . . .	v
FIGURES . . . . .	xlviii
SECTION I DIVWAG SYSTEM GENERAL DESCRIPTION . . . . .	I-1
Chapter 1. INTRODUCTION . . . . .	I-1-1
1. Purpose . . . . .	I-1-1
2. Scope . . . . .	I-1-1
3. Organization . . . . .	I-1-1
Chapter 2. MODEL DESIGN AND DEVELOPMENT . . . . .	I-2-1
1. Need for a Division War Game Model . . . . .	I-2-1
2. Division War Game Model Development . . . . .	I-2-2
Chapter 3. SUMMARY DESCRIPTION . . . . .	I-3-1
1. Model Objective . . . . .	I-3-1
2. Model Capabilities . . . . .	I-3-1
3. Functional Components and Capabilities . . . . .	I-3-2
Chapter 4. COMPUTER SYSTEM REQUIREMENTS . . . . .	I-4-1
Appendix A. DEFINITIONS OF TERMS . . . . .	I-A-1
SECTION II CONSTANT DATA INPUT PROCESSOR . . . . .	II-1
Chapter 1. GENERAL DESCRIPTION OF THE CONSTANT DATA INPUT PROCESSOR . . . . .	II-1-1
1. Introduction . . . . .	II-1-1
2. Significance of this Section . . . . .	II-1-1
3. Organization of this Section . . . . .	II-1-1
4. Sequence of Load Programs . . . . .	II-1-3
Chapter 2. EXECUTIVE CONTROL DATA INPUT . . . . .	II-2-1
Appendix A. Input Requirements for Executive Control Data Input . . . . .	II-2-A-1

## CONTENTS (Continued)

	<u>Page</u>
Appendix B. Executive Control Data Input Program Descriptions	II-2-B-1
1. Introduction . . . . .	II-2-B-1
2. Routine PREDIV . . . . .	II-2-B-1
3. Routine OVRLY7 . . . . .	II-2-B-8
4. Routine OVLY10 . . . . .	II-2-B-8
Appendix C. Output Descriptions for Executive Control Data Input . . . . .	II-2-C-1
Appendix D. Source Listings for Constant Data Input Processor Executive Control . . . . .	II-2-D-1
Chapter 3. ORGANIZATION AND EQUIPMENT DATA INPUT PROGRAMS . . . .	II-3-1
1. Introduction . . . . .	II-3-1
2. Routines . . . . .	II-3-1
3. Files . . . . .	II-3-1
Appendix A. Organization and Equipment Data Load Input Requirements . . . . .	II-3-A-1
Appendix B. Organization and Equipment Data Input Program Descriptions . . . . .	II-3-B-1
1. Introduction . . . . .	II-3-B-1
2. Routine LDTOE . . . . .	II-3-B-1
3. Routine SRCH . . . . .	II-3-B-14
4. Routine DMPTOE . . . . .	II-3-B-16
5. Routine SORT . . . . .	II-3-B-20
Appendix C. Organization and Equipment Data Load Output Descriptions . . . . .	II-3-C-1
1. Introduction . . . . .	II-3-C-1
2. Eighty-Column Card Image of Input Data . . . . .	II-3-C-1
3. Secondary Equipment for Blue and Red . . . . .	II-3-C-1
4. Unit Type Designator Breakdown Table . . . . .	II-3-C-4
5. Unit Type Designator Directory . . . . .	II-3-C-4
6. Authorized Personnel and Equipment . . . . .	II-3-C-4
Appendix D. Source Listings for Constant Data Input Processor Organization and Equipment . . . . .	II-3-D-1
Chapter 4. TASK ORGANIZATION DATA INPUT PROGRAMS . . . . .	II-4-1
1. Introduction . . . . .	II-4-1
2. Routines . . . . .	II-4-1
3. Files . . . . .	II-4-2

## CONTENTS (Continued)

	<u>Page</u>
Appendix A. Task Organization Data Load Input Requirements . .	II-4-A-1
Appendix B. Task Organization Data Input Program Descriptions	II-4-B-1
1. Introduction . . . . .	II-4-B-1
2. Routine ECHLON . . . . .	II-4-B-2
3. Routine RECOG1 . . . . .	II-4-B-8
4. Routine COMMCK . . . . .	II-4-B-16
5. Routine COMPRS . . . . .	II-4-B-16
6. Routine CFIND . . . . .	II-4-B-18
7. Routine BLNKOT . . . . .	II-4-B-20
8. Routine CORDIN . . . . .	II-4-B-20
9. Routine SUPORT . . . . .	II-4-B-22
10. Routine SEQUEN . . . . .	II-4-B-25
11. Routine ADDUNT . . . . .	II-4-B-31
12. Routine TOEPUT . . . . .	II-4-B-31
13. Routine UPDATE . . . . .	II-4-B-41
14. Routine CRAT31 . . . . .	II-4-B-49
15. Routine RELATR . . . . .	II-4-B-50
16. Routine SUPPLY . . . . .	II-4-B-54
17. Routine TALLY . . . . .	II-4-B-60
Appendix C. Task Organization Data Load Output Descriptions .	II-4-C-1
1. Introduction . . . . .	II-4-C-1
2. Unit Type Designator Directory . . . . .	II-4-C-1
3. Card Image . . . . .	II-4-C-1
4. RECOGN1 Table . . . . .	II-4-C-1
5. Message . . . . .	II-4-C-1
6. Supply Status Table . . . . .	II-4-C-1
7. Unit Resolution List. . . . .	II-4-C-1
8. Unit Type Designator Summary . . . . .	II-4-C-7
9. Units Created . . . . .	II-4-C-7
Appendix D. Source Listings for Constant Data Input Processor Task Organization . . . . .	II-4-D-1
Chapter 5. ENVIRONMENTAL DATA INPUT PROGRAMS . . . . .	II-5-1
1. Introduction . . . . .	II-5-1
2. Routines . . . . .	II-5-1
3. Files . . . . .	II-5-1
Appendix A. Environmental Data Load Input Requirements . . . .	II-5-A-1
Appendix B. Environmental Data Input Program Descriptions . .	II-5-B-1
1. Introduction . . . . .	II-5-B-1
2. Routine ELEVLD. . . . .	II-5-B-1

CONTENTS (Continued)

	<u>Page</u>
3. Routine TERAIn . . . . .	II-5-B-7
4. Routine TERNLD . . . . .	II-5-B-7
5. Routine TERNDP . . . . .	II-5-B-11
6. Routine WETHLD . . . . .	II-5-B-12
Appendix C. Environmental Data Load Output Descriptions . . .	II-5-C-1
Appendix D Source Listings for Constant Data Input Processor Environmental Representation . . . . .	II-5-D-1
Chapter 6. UNIT GEOMETRY DATA INPUT PROGRAMS . . . . .	II-6-1
1. Introduction . . . . .	II-6-1
2. Routines . . . . .	II-6-1
3. Files . . . . .	II-6-1
Appendix A. Unit Geometry Data Load Input Requirements . . . .	II-6-A-1
Appendix B. Unit Geometry Data Input Program Descriptions	II-6-B-1
1. Introduction . . . . .	II-6-B-1
2. Routine DIMLD . . . . .	II-6-B-1
3. Routine LOAD28 . . . . .	II-6-B-1
4. Routine DUMP28 . . . . .	II-6-B-10
Appendix C. Unit Geometry Data Load Output Descriptions . . .	II-6-C-1
1. Introduction . . . . .	II-6-C-1
2. Eighty-Column Card Image, Data File 28 . . . . .	II-6-C-1
3. Unit Type Designator Record Number Directory . . . . .	II-6-C-1
4. Unit Descriptions . . . . .	II-6-C-1
Appendix D. Source Listings for Constant Data Input Processor Unit Geometry . . . . .	II-6-D-1
Chapter 7. INTELLIGENCE AND CONTROL DATA INPUT PROGRAMS . . . . .	II-7-1
1. Introduction . . . . .	II-7-1
2. Routines . . . . .	II-7-1
3. Files . . . . .	II-7-1
Appendix A. Intelligence and Control Data Load Input Requirements . . . . .	II-7-A-1
Appendix B. Intelligence and Control Data Input Program Descriptions . . . . .	II-7-B-1
1. Introduction . . . . .	II-7-B-1

## CONTENTS (Continued)

	<u>Page</u>
2. Routine INCSLD . . . . .	II-7-B-1
3. Routine SET20 . . . . .	II-7-B-10
Appendix C. Intelligence and Control Data Load Output Descriptions . . . . .	II-7-C-1
1. Introduction . . . . .	II-7-C-1
2. Eighty-Column Card Image . . . . .	II-7-C-2
3. Constant Data on Sensors . . . . .	II-7-C-2
4. Unattended Ground Sensors . . . . .	II-7-C-2
5. Equipment Item Code for Target Category . . . . .	II-7-C-2
6. Standing Operating Procedure for Reporting Hostile Forces . . . . .	II-7-C-2
7. Flow Delay Times . . . . .	II-7-C-2
8. Processing and Decision Delay Times . . . . .	II-7-C-2
9. Unit Size Based on Personnel and Equipment Counts . . . . .	II-7-C-2
10. Target Inferences . . . . .	II-7-C-11
Appendix D. Source Listings for Constant Data Input Processor Intelligence and Control . . . . .	II-7-D-1
Chapter 8. GROUND COMBAT DATA INPUT PROGRAMS . . . . .	II-8-1
1. Introduction . . . . .	II-8-1
2. Routines . . . . .	II-8-1
3. Files . . . . .	II-8-1
Appendix A. Ground Combat Data Load Input Requirements . . . . .	II-8-A-1
Appendix B. Ground Combat Data Input Program Descriptions . . . . .	II-8-B-1
1. Introduction . . . . .	II-8-B-1
2. Routine GCMLD . . . . .	II-8-B-1
3. Routine AROTE . . . . .	II-8-B-6
4. Routine LINEAR . . . . .	II-8-B-6
5. Routine SPP . . . . .	II-8-B-7
6. Routine FLIP . . . . .	II-8-B-8
7. Routine DUMP39 . . . . .	II-8-B-8
Appendix C. Ground Combat Data Load Output Descriptions . . . . .	II-8-C-1
1. Introduction . . . . .	II-8-C-1
2. Eighty-Column Card Image . . . . .	II-8-C-1
3. Line of Sight . . . . .	II-8-C-1
4. Target and Pinpoint Probabilities . . . . .	II-8-C-4
5. Weapon System Specifications . . . . .	II-8-C-4
6. Weapon Target Kill Probability . . . . .	II-8-C-4
7. Sensor Distribution and Performance . . . . .	II-8-C-8

## CONTENTS (Continued)

	<u>Page</u>
Appendix D. Source Listings for Constant Data Input Processor Ground Combat . . . . .	II-8-D-1
Chapter 9. AREA FIRE DATA INPUT PROGRAMS . . . . .	II-9-1
1. Introduction . . . . .	II-9-1
2. Routines . . . . .	II-9-1
3. Files . . . . .	II-9-1
Appendix A. Area Fire Data Load Input Requirements . . . . .	II-9-A-1
Appendix B. Area Fire Data Input Program Descriptions . . . . .	II-9-B-1
1. Introduction . . . . .	II-9-B-1
2. Routine AFMLD . . . . .	II-9-B-1
3. Routine LOAD29 . . . . .	II-9-B-3
4. Routine LOAD30 . . . . .	II-9-B-12
5. Routine DUMP29 . . . . .	II-9-B-20
6. Routine DUMP30 . . . . .	II-9-B-23
Appendix C. Area Fire Data Load Output Descriptions . . . . .	II-9-C-1
1. Introduction . . . . .	II-9-C-1
2. Data File 29 80-Column Card Image . . . . .	II-9-C-1
3. Weapons and Munitions Characteristics and Performance . . . . .	II-9-C-1
4. Weapons and Munitions Delivery Parameters . . . . .	II-9-C-1
5. Weapons and Munitions Description . . . . .	II-9-C-5
6. Eighty-Column Card Image, Personnel Data . . . . .	II-9-C-5
7. Personnel Data . . . . .	II-9-C-5
Appendix D. Source Listings for Constant Data Input Processor Area Fire . . . . .	II-9-D-1
Chapter 10. TACFIRE DATA INPUT PROGRAMS . . . . .	II-10-1
1. Introduction . . . . .	II-10-1
2. Routines . . . . .	II-10-1
3. Files . . . . .	II-10-1
Appendix A. TACFIRE Data Load Input Requirements . . . . .	II-10-A-1
Appendix B. TACFIRE Data Input Program Descriptions . . . . .	II-10-B-1
1. Introduction . . . . .	II-10-B-1
2. Routine TACLD . . . . .	II-10-B-1
3. Routine LOAD25 . . . . .	II-10-B-1
4. Routine OSORT . . . . .	II-10-B-13
5. Routine DUMP25 . . . . .	II-10-B-14

## CONTENTS (Continued)

	<u>Page</u>
Appendix C. TACFIRE Data Load Output Descriptions . . . . .	II-10-C-1
1. Introduction . . . . .	II-10-C-1
2. Data File 25 Data Card Image . . . . .	II-10-C-1
3. Priority and Method of Attack Tables . . . . .	II-10-C-1
4. Summary Report of Target Priorities . . . . .	II-10-C-1
Appendix D. Source Listings for Constant Data Input Processor TACFIRE . . . . .	II-10-D-1
Chapter 11. AIR GROUND ENGAGEMENT DATA INPUT PROGRAMS . . . . .	II-11-1
1. Introduction . . . . .	II-11-1
2. Routines . . . . .	II-11-1
3. Files . . . . .	II-11-1
Appendix A. Air Ground Engagement Data Load Input Requirements	II-11-A-1
Appendix B. Air Ground Engagement Data Input Program Descriptions . . . . .	II-11-B-1
1. Introduction . . . . .	II-11-B-1
2. Routine AIRLD . . . . .	II-11-B-1
Appendix C. Air Ground Engagement Data Load Output Descriptions . . . . .	II-11-C-1
1. Introduction . . . . .	II-11-C-1
2. Data File 26 80-Column Card Image . . . . .	II 11-C-1
3. Aircraft and Munitions Minimum and Maximum Requirements	II-11-C-1
4. Aircraft Preparation Delay Times . . . . .	II-11-C-1
5. Aircraft Engagement Parameters . . . . .	II-11-C-1
6. Aircraft Minimum Abort Data . . . . .	II-11-C-1
7. Aircraft Landing Time . . . . .	II-11-C-8
8. Aircraft Flight Speeds . . . . .	II-11-C-8
9. Aircraft New Mission Availability Time . . . . .	II-11-C-8
10. Aircraft Degradation for Maintenance . . . . .	II-11-C-8
11. Aircraft Vulnerable Areas . . . . .	II-11-C-8
12. Air Defense Weapon Characteristics . . . . .	II-11-C-8
13. Air Defense Weapons Effectiveness . . . . .	II-11-C-8
14. Air Defense Weapon Degradation from Terrain and Vegetation . . . . .	II-11-C-8
15. Engagement Results . . . . .	II-11-C-17
Appendix D. Source Listings for Constant Data Input Processor Air Ground Engagement . . . . .	II-11-D-1



## CONTENTS (Continued)

	<u>Page</u>
Chapter 12. SUPPRESSION DATA INPUT PROGRAMS . . . . .	II-12-1
1. Introduction . . . . .	II-12-1
2. Routines . . . . .	II-12-1
3. Files . . . . .	II-12-1
Appendix A. Suppression Data Load Input Requirements . . . . .	II-12-A-1
Appendix B. Suppression Data Input Program Descriptions . . . . .	II-12-B-1
1. Introduction . . . . .	II-12-B-1
2. Routine SPRSLD . . . . .	II-12-B-1
3. Routine SUPDMP . . . . .	II-12-B-1
Appendix C. Suppression Data Load Output Descriptions . . . . .	II-12-C-1
Appendix D. Source Listings for Constant Data Input Processor Suppression . . . . .	II-12-D-1
Chapter 13. NUCLEAR ASSESSMENT DATA INPUT PROGRAMS . . . . .	II-13-1
1. Introduction . . . . .	II-13-1
2. Routines . . . . .	II-13-1
3. Files . . . . .	II-13-1
Appendix A. Nuclear Assessment Data Load Input Requirements . . . . .	II-13-A-1
Appendix B. Nuclear Assessment Data Input Program Descriptions . . . . .	II-13-B-1
1. Introduction . . . . .	II-13-B-1
2. Routine NUCLD . . . . .	II-13-B-1
3. Routine FLD30 . . . . .	II-13-B-3
4. Routine CKARY . . . . .	II-13-B-9
5. Routine DMP30 . . . . .	II-13-B-9
Appendix C. Nuclear Assessment Data Load Output Descriptions . . . . .	II-13-C-1
1. Introduction . . . . .	II-13-C-1
2. Nuclear Assessment 80-Column Card Image . . . . .	II-13-C-1
3. Weapon Combination Names . . . . .	II-13-C-3
4. Weapon Statistics . . . . .	II-13-C-3
5. Fuze Table Entries . . . . .	II-13-C-3
6. Yield Table . . . . .	II-13-C-3
7. EOH Links . . . . .	II-13-C-3
8. EOH Data . . . . .	II-13-C-3
9. Yield Codes and Barrier Data . . . . .	II-13-C-3
10. Unprotected Personnel Distribution . . . . .	II-13-C-3
11. Special Factors for Both Teams . . . . .	II-13-C-3

CONTENTS (Continued)

	<u>Page</u>
Appendix D. Source Listings for Constant Data Input Processor Nuclear Assessment . . . . .	II-13-D-1
Chapter 14. MOVEMENT DATA INPUT PROGRAMS . . . . .	II-14-1
1. Introduction . . . . .	II-14-1
2. Routines . . . . .	II-14-1
3. Files . . . . .	II-14-1
Appendix A. Movement Data Load Input Requirements . . . . .	II-14-A-1
Appendix B. Movement Data Input Program Descriptions . . . . .	II-14-B-1
1. Introduction . . . . .	II-14-B-1
2. Routine MOVELD . . . . .	II-14-B-1
3. Routine DPMOVE . . . . .	II-14-B-5
Appendix C. Movement Data Load Output Descriptions . . . . .	II-14-C-1
1. Introduction . . . . .	II-14-C-1
2. Eighty-Column Card Image . . . . .	II-14-C-1
3. Mobility Category and Travel Mode . . . . .	II-14-C-2
4. Mobility Class and Exclusion Table . . . . .	II-14-C-2
5. Unit Road and Cross Country Movement . . . . .	II-14-C-2
6. Equipment Cross Country and Road Movement Rates . . . . .	II-14-C-2
7. Ground and Air Fuel Consumption Rates . . . . .	II-14-C-2
Appendix D. Source Listings for Constant Data Input Processor Movement . . . . .	II-14-D-1
Chapter 15. ENGINEER DATA INPUT PROGRAMS . . . . .	II-15-1
1. Introduction . . . . .	II-15-1
2. Routines . . . . .	II-15-1
3. Files . . . . .	II-15-1
Appendix A. Engineer Data Load Input Requirements . . . . .	II-15-A-1
Appendix B. Engineer Data Input Program Descriptions . . . . .	II-15-B-1
1. Introduction . . . . .	II-15-B-1
2. Routines ENGLD and DUMP17 . . . . .	II-15-B-1
3. Routines GET17, IFILE, IDENT, and ITYPCK . . . . .	II-15-B-8
4. Routine BARLD . . . . .	II-15-B-8
Appendix C. Engineer Data Load Output Descriptions . . . . .	II-15-C-1
1. Introduction . . . . .	II-15-C-1
2. Barrier Segment Location and Status . . . . .	II-15-C-1

## CONTENTS (Continued)

	<u>Page</u>
3. Engineer 80-Column Card Image . . . . .	II-15-C-1
4. Engineering Task Parameters . . . . .	II-15-C-1
5. Environmental and Troop Rates and Rate Modifiers . . . . .	II-15-C-1
6. Equipment Requirements and Rate Modifiers . . . . .	II-15-C-1
7. Minefield Forcing Casualties . . . . .	II-15-C-1
Appendix D. Source Listings for Constant Data Input Processor Engineer . . . . .	II-15-D-1
Chapter 16. AIRMOBILE DATA INPUT PROGRAMS . . . . .	II-16-1
1. Introduction . . . . .	II-16-1
2. Routines . . . . .	II-16-1
3. Files . . . . .	II-16-1
Appendix A. Airmobile Data Load Input Requirements . . . . .	II-16-A-1
Appendix B. Airmobile Data Input Program Descriptions . . . . .	II-16-B-1
1. Introduction . . . . .	II-16-B-1
2. Routine FIL7LD . . . . .	II-16-B-1
Appendix C. Output Descriptions for Airmobile Data Input . . . . .	II-16-C-1
Appendix D. Source Listings for Constant Data Input Processor Airmobile . . . . .	II-16-D-1
Chapter 17. COMBAT SERVICE SUPPORT DATA INPUT PROGRAMS . . . . .	II-17-1
1. Introduction . . . . .	II-17-1
2. Routines . . . . .	II-17-1
3. Files . . . . .	II-17-1
Appendix A. Combat Service Support Data Load Input Requirements . . . . .	II-17-A-1
Appendix B. Combat Service Support Data Input Program Descriptions . . . . .	II-17-B-1
1. Introduction . . . . .	II-17-B-1
2. Routine CSSLD . . . . .	II-17-B-1
3. Routine LOAD11 . . . . .	II-17-B-1
4. Routine EDITF1 . . . . .	II-17-B-8
5. Routine EDITF2 . . . . .	II-17-B-9
6. Routine DUMP11 . . . . .	II-17-B-9

## CONTENTS (Continued)

	<u>Page</u>
Appendix C. Combat Service Support Data Load Output	
Descriptions . . . . .	II-17-C-1
1. Introduction . . . . .	II-17-C-1
2. Eighty-Column Card Image . . . . .	II-17-C-1
3. Elapsed Time to Handle Vehicles and Class I Consumption Rate . . . . .	II-17-C-1
4. Weights and Volumes of Consumables and Vehicles . . . . .	II-17-C-4
5. Classes of Supply . . . . .	II-17-C-4
6. Transport Preference for Hauling Consumables . . . . .	II-17-C-4
7. Major End Items and Personnel Available Daily for Resupply . . . . .	II-17-C-4
Appendix D. Source Listings for Constant Data Input Processor Combat Service Support . . . . .	II-17-D-1
SECTION III ORDERS INPUT PROCESSOR . . . . .	III-1
Chapter 1. GENERAL DESCRIPTION OF THE ORDERS INPUT PROCESSOR . . . . .	III-1-1
1. Introduction . . . . .	III-1-1
2. Input Processor Subsystems . . . . .	III-1-1
Chapter 2. THE DIVWAG SCENARIO LANGUAGE COMPILER . . . . .	III-2-1
1. General Description . . . . .	III-2-1
2. Description of Processing . . . . .	III-2-1
3. DSL Logical Flow . . . . .	III-2-10
4. Routine Descriptions . . . . .	III-2-10
Appendix A. Input Requirements for DIVWAG Scenario Language Compiler . . . . .	III-2-A-1
1. Introduction . . . . .	III-2-A-1
2. DSL Composition . . . . .	III-2-A-1
3. DSL Order Clauses . . . . .	III-2-A-6
4. DSL Conditional Clauses . . . . .	III-2-A-25
5. Control Cards and Deck Structure . . . . .	III-2-A-30
6. DSL Rules and Techniques . . . . .	III-2-A-33
7. DIVWAG Scenario Language Vocabulary . . . . .	III-2-A-38
Appendix B. DIVWAG Scenario Language Compiler Program Descriptions . . . . .	III-2-B-1
1. Introduction . . . . .	III-2-B-1
2. Common DSLMN . . . . .	III-2-B-1
3. Common CONST . . . . .	III-2-B-3

## CONTENTS (Continued)

	<u>Page</u>
Appendix A. Operating Instructions Data Input Requirements . .	III-3-A-1
1. Sensor Control . . . . .	III-3-A-1
2. Introduction of Intelligence . . . . .	III-3-A-13
3. Fire Support Allocation Control . . . . .	III-3-A-18
4. OPERINS Data Deck Structure . . . . .	III-3-A-21
Appendix B. Operating Instructions Program Descriptions . . .	III-3-B-1
1. Introduction . . . . .	III-3-B-1
2. Routine DATAST . . . . .	III-3-B-1
3. Routine VINKLD . . . . .	III-3-B-4
4. Routine CNVRT . . . . .	III-3-B-11
5. Routine AREA . . . . .	III-3-B-12
6. Routine LDASR . . . . .	III-3-B-12
7. Routine PLOAD . . . . .	III-3-B-16
8. Routine SLOAD . . . . .	III-3-B-16
9. Routine INTLIN . . . . .	III-3-B-17
10. Routines IUIDF, UTSR, and EVTSET . . . . .	III-3-B-24
Appendix C. Output Descriptions for Operating Instructions . .	III-3-C-1
Appendix D. Source Listings for Orders Input Processor	
Operating Instructions . . . . .	III-3-D-1

## CONTENTS (Continued)

### PART 2

	<u>Page</u>
TITLE PAGE . . . . .	i
CONTENTS . . . . .	iii
FIGURES . . . . .	ix
SECTION IV PERIOD PROCESSOR . . . . .	IV-1
Chapter 1. GENERAL DESCRIPTION OF THE PERIOD PROCESSOR . . . . .	IV-1-1
1. Introduction . . . . .	IV-1-1
2. Processing Sequence . . . . .	IV-1-1
3. DIVWAG System Models . . . . .	IV-1-6
Chapter 2. EXECUTIVE CONTROL ROUTINES . . . . .	IV-2-1
1. Introduction . . . . .	IV-2-1
2. Primary Executive . . . . .	IV-2-1
3. Event Scheduling Routines . . . . .	IV-2-1
4. Event Table . . . . .	IV-2-1
5. Battle Update . . . . .	IV-2-4
Appendix A. Executive Control Input Requirements . . . . .	IV-2-A-1
Appendix B. Executive Control Program Descriptions . . . . .	IV-2-B-1
1. Introduction . . . . .	IV-2-B-1
2. Routine MAIN . . . . .	IV-2-B-1
3. Routine MINUET . . . . .	IV-2-B-1
4. Routine ESR . . . . .	IV-2-B-15
5. Routine ARCNRL . . . . .	IV-2-B-29
6. Routine FORCRD . . . . .	IV-2-B-38
7. Routine ENGORD . . . . .	IV-2-B-38
8. Routine EVTSET . . . . .	IV-2-B-40
9. Routine PUTEVT . . . . .	IV-2-B-42
10. Routine GETEVT . . . . .	IV-2-B-45
11. Routine INRUPT . . . . .	IV-2-B-45
Appendix C. Output Descriptions for Executive Control Routines . . . . .	IV-2-C-1
Appendix D. Source Listings for Period Processor Executive Control . . . . .	IV-2-D-1
Chapter 3. UNIT REPRESENTATION . . . . .	IV-3-1
1. General . . . . .	IV-3-1

## CONTENTS (Continued)

	<u>Page</u>
2. Task Organization . . . . .	IV-3-3
Appendix A. Unit Representation Input Requirements . . . . .	IV-3-A-1
1. Introduction . . . . .	IV-3-A-1
2. Organization and Equipment Constant Data Preparation . . . . .	IV-3-A-1
3. DIVWAG Submodel Operations Using TOE Data . . . . .	IV-3-A-1
4. Considerations Prior to Preparing TOE Data . . . . .	IV-3-A-2
5. Preliminary Steps . . . . .	IV-3-A-4
6. Senior and Subordinate Type Units . . . . .	IV-3-A-8
7. Secondary Equipment List . . . . .	IV-3-A-15
8. Table of Organization and Equipment . . . . .	IV-3-A-17
9. Bulk-loaded Expendable Supplies . . . . .	IV-3-A-20
10. TOE Data Deck Structure . . . . .	IV-3-A-22
11. Task Organization Constant Data Preparation . . . . .	IV-3-A-25
12. Submodel Use of Data . . . . .	IV-3-A-25
13. Considerations Prior to Preparing Data . . . . .	IV-3-A-26
14. Basic Organization Input . . . . .	IV-3-A-31
15. Supply Point Data . . . . .	IV-3-A-39
16. Data Deck Setup . . . . .	IV-3-A-42
Appendix B. Unit Representation Program Descriptions . . . . .	IV-3-B-1
1. Introduction . . . . .	IV-3-B-1
2. Routine TRFR . . . . .	IV-3-B-1
3. Routine ASSIGN . . . . .	IV-3-B-4
4. Routine ASSUME . . . . .	IV-3-B-4
5. Routine DETACH . . . . .	IV-3-B-9
6. Routine DTCH31 . . . . .	IV-3-B-16
7. Routine JOIN . . . . .	IV-3-B-21
8. Routine BIAS . . . . .	IV-3-B-26
9. Routine JOIN31 . . . . .	IV-3-B-27
Appendix C. Output Descriptions for Unit Representation . . . . .	IV-3-C-1
Appendix D. Source Listing for Period Processor Unit Representation . . . . .	IV-3-D-1
Chapter 4. ENVIRONMENTAL REPRESENTATION . . . . .	IV-4-1
1. Introduction . . . . .	IV-4-1
2. Terrain Representation . . . . .	IV-4-1
3. Weather . . . . .	IV-4-8
4. Terrain Masking . . . . .	IV-4-9
Appendix A. Environmental Representation Input Requirements . . . . .	IV-4-A-1
1. Introduction . . . . .	IV-4-A-1

## CONTENTS (Continued)

	<u>Page</u>
2. Data Preparation for the Terrain File . . . . .	IV-4-A-1
3. Terrain Elevation Data . . . . .	IV-4-A-8
4. Data Preparation for Weather Selection . . . . .	IV-4-A-11
Appendix B. Environmental Representation Program Descriptions	IV-4-B-1
1. Introduction . . . . .	IV-4-B-1
2. Routine UPMASK . . . . .	IV-4-B-1
3. Routine ELEVAT . . . . .	IV-4-B-3
4. Routine WETTHR . . . . .	IV-4-B-5
Appendix C. Environmental Representation Output Descriptions .	IV-4-C-1
1. Introduction . . . . .	IV-4-C-1
2. Routine UPMASK . . . . .	IV-4-C-1
3. Routine WETTHR . . . . .	IV-4-C-1
Appendix D. Source Listings for Period Processor Environmental Representation . . . . .	IV-4-D-1
Chapter 5. BATTLEFIELD AND UNIT GEOMETRY . . . . .	IV-5-1
1. Introduction . . . . .	IV-5-1
2. Battlefield Orientation . . . . .	IV-5-1
3. Representation of Resolution Units . . . . .	IV-5-5
Appendix A. Unit Geometry Input Requirements . . . . .	IV-5-A-1
1. Introduction . . . . .	IV-5-A-1
2. Planning Matrix . . . . .	IV-5-A-1
3. Units with Common Dimensions and Distributions . . . . .	IV-5-A-4
4. Equipment Issued to UTDs That Have Been Combined . . . . .	IV-5-A-8
5. Unit Geometry Card . . . . .	IV-5-A-10
6. Distribution of Personnel and Equipment . . . . .	IV-5-A-12
7. Unit Geometry Data Deck Structure . . . . .	IV-5-A-14
Appendix B. Battlefield and Unit Geometry Program Descriptions	IV-5-B-1
1. Introduction . . . . .	IV-5-B-1
2. Routine ZONE . . . . .	IV-5-B-1
3. Routine AFLAG . . . . .	IV-5-B-4
4. Routine UPDVAL . . . . .	IV-5-B-6
Appendix C. Output Descriptions for Battlefield and Unit Geometry Routines . . . . .	IV-5-C-1
1. Introduction . . . . .	IV-5-C-1



## CONTENTS (Continued)

	<u>Page</u>
2. Routine ZONE . . . . .	IV-5-C-1
3. Routine UPDVUL . . . . .	IV-5-C-1
Appendix D. Source Listings for Period Processor Battlefield and Unit Geometry . . . . .	IV-5-D-1
Chapter 6. INTELLIGENCE AND CONTROL MODEL . . . . .	IV-6-1
1. Military Activities Represented . . . . .	IV-6-1
2. Design of the Intelligence and Control Model . . . . .	IV-6-7
3. Intelligence and Control Model Interaction with Other DIVWAG Models . . . . .	IV-6-10
4. Collection Submodel . . . . .	IV-6-11
5. Creative Processing Submodel . . . . .	IV-6-67
6. Decision Submodel . . . . .	IV-6-74
7. Assessment of Ground Sensors . . . . .	IV-6-83
Appendix A. Intelligence and Control Model Input Requirements . . . . .	IV-6-A-1
1. Introduction . . . . .	IV-6-A-1
2. Intelligence and Control Input Card Format . . . . .	IV-6-A-1
3. General Radar Data . . . . .	IV-6-A-2
4. General Radar Data, Continued . . . . .	IV-6-A-6
5. Radar Ranges . . . . .	IV-6-A-9
6. General Countermortar/Counterbattery Radar Data . . . . .	IV-6-A-13
7. General Countermortar/Counterbattery Radar Data, Continued . . . . .	IV-6-A-15
8. Ground Based Radar Range Brackets for Sensor Movement . . . . .	IV-6-A-17
9. Air Reconnaissance/Surveillance Constant Data Requirements . . . . .	IV-6-A-19
10. Sensor Constant Data Input Format . . . . .	IV-6-A-25
11. Sensor Combination Load Input Format . . . . .	IV-6-A-30
12. LOH Decision Control Matrix . . . . .	IV-6-A-30
13. LOH or Army Aviation Fixed . . . . . Wing Reconnaissance Vehicle TOE . . . . .	IV-6-A-35
14. LOH or Army Aviation Fixed Wing Reconnaissance Vehicle TOE Card Format . . . . .	IV-6-A-35
15. Intelligence Handling Elapsed Time . . . . .	IV-6-A-38
16. Target Inferences . . . . .	IV-6-A-40
17. Target Inferences from Counts and Intervals, Continued . . . . .	IV-6-A-47
18. Dissemination of Intelligence Information . . . . .	IV-6-A-51
19. Unattended Ground Sensors . . . . .	IV-6-A-57
20. Coding of Vehicles and Weapons . . . . .	IV-6-A-60
21. Intelligence and Control Data Deck Structure . . . . .	IV-6-A-62
Appendix B. Intelligence and Control Model Program Descriptions . . . . .	IV-6-B-1

## CONTENTS (Continued)

	<u>Page</u>
1. Introduction . . . . .	IV-6-B-1
2. Routine INCEXC . . . . .	IV-6-B-1
3. Routine CCOLLM . . . . .	IV-6-B-4
4. Routine COLLECT . . . . .	IV-6-B-14
5. Routine CCOLLF . . . . .	IV-6-B-26
6. Routine WORKIT . . . . .	IV-6-B-37
7. Routine PLNINT . . . . .	IV-6-B-42
8. Routine CIRINT . . . . .	IV-6-B-45
9. Routine CKINT . . . . .	IV-6-B-48
10. Routine IQANGC . . . . .	IV-6-B-50
11. Routine CPDT . . . . .	IV-6-B-52
12. Routine PROCES . . . . .	IV-6-B-57
13. Routine UTSR . . . . .	IV-6-B-70
14. Routine SUTR . . . . .	IV-6-B-73
15. Routine TAR . . . . .	IV-6-B-76
16. Routine ARESPO . . . . .	IV-6-B-79
17. Routine DECIDT . . . . .	IV-6-B-79
18. Routine DECIDE . . . . .	IV-6-B-81
19. Routine MOVSEN . . . . .	IV-6-B-97
20. Routine SASSMT . . . . .	IV-6-B-102
21. Routine SENSRH . . . . .	IV-6-B-110
22. Routine UPSENR . . . . .	IV-6-B-111
23. Reconnaissance Overlay Routines . . . . .	IV-6-B-113
24. Routine RCMAN . . . . .	IV-6-B-115
25. Routine ITGTX . . . . .	IV-6-B-118
26. Routine NSUBS . . . . .	IV-6-B-118
27. Routine KYSUBU . . . . .	IV-6-B-119
28. Routine NUMTGT . . . . .	IV-6-B-119
29. Routine RECONN . . . . .	IV-6-B-126
30. Routine RCONDT . . . . .	IV-6-B-128
31. Routine GETSTS . . . . .	IV-6-B-138
32. Routine GETSTD . . . . .	IV-6-B-138
33. Routine ISTREF . . . . .	IV-6-B-141
34. Routine MOVREC . . . . .	IV-6-B-141
35. Routine RECHOM . . . . .	IV-6-B-147
36. Routine RECEND . . . . .	IV-6-B-153
37. Routine RECDMP . . . . .	IV-6-B-155
38. Routine IRCSRC . . . . .	IV-6-B-158
39. Routine SHTDWN . . . . .	IV-6-B-159
40. Routine RECON1 . . . . .	IV-6-B-159
41. Routine RECON2 . . . . .	IV-6-B-182
42. Routine RECON3 . . . . .	IV-6-B-194
43. Routine RECN3B . . . . .	IV-6-B-214
44. Routine NONRES . . . . .	IV-6-B-224
Appendix C. Output Descriptions for Intelligence and Control	
Model . . . . .	IV-6-C-1
1. Introduction . . . . .	IV-6-C-1

## CONTENTS (Continued)

	<u>Page</u>
2. FILE12 and Sensor Redundancy . . . . .	IV-6-C-1
3. Countermortar/Counterbattery Radar . . . . .	IV-6-C-3
4. MOVSEN . . . . .	IV-6-C-6
5. Sensor Assessments . . . . .	IV-6-C-8
6. Routine RCONDT . . . . .	IV-6-C-11
7. Routine MOVREC . . . . .	IV-6-C-11
8. Routines GETSTD, SHTDWN, RECHOM, and RECDMP . . . . .	IV-6-C-11
9. Routine RECON1 . . . . .	IV-6-C-15
10. Routine RECON2 . . . . .	IV-6-C-15
11. Routine RECON3 . . . . .	IV-6-C-21
Appendix D. Source Listings for Period Processor Intelligence and Control Model . . . . .	IV-6-D-1
Appendix E. References . . . . .	IV-6-E-1

## CONTENTS (Continued)

### PART 3

	<u>Page</u>
TITLE PAGE . . . . .	i
CONTENTS . . . . .	iii
FIGURES . . . . .	ix
SECTION IV PERIOD PROCESSOR (Continued)	
Chapter 7. GROUND COMBAT MODEL . . . . .	IV-7-1
1. Military Activity Represented . . . . .	IV-7-1
2. Model Design . . . . .	IV-7-2
3. Submodel Specifications . . . . .	IV-7-12
4. History Tape Output . . . . .	IV-7-73
Appendix A. Ground Combat Model Input Requirements . . . . .	IV-7-A-1
1. Introduction . . . . .	IV-7-A-1
2. Weapon/Transport System . . . . .	IV-7-A-2
3. Specifications for the Weapon System . . . . .	IV-7-A-6
4. Specifications for Sensors . . . . .	IV-7-A-10
5. Targets and their Priorities . . . . .	IV-7-A-15
6. Line of Sight . . . . .	IV-7-A-18
7. Horizon and Background . . . . .	IV-7-A-22
8. Ground Combat Data Structure . . . . .	IV-7-A-25
Appendix B. Ground Combat Model Program Descriptions . . . . .	IV-7-B-1
1. Introduction . . . . .	IV-7-B-1
2. Routine GCMDT . . . . .	IV-7-B-2
3. Routine GCDRVR . . . . .	IV-7-B-5
4. Routine GCM . . . . .	IV-7-B-6
5. Routine GSETUP . . . . .	IV-7-B-18
6. Routine DNUMBER . . . . .	IV-7-B-27
7. Routine STRBAR . . . . .	IV-7-B-29
8. Routine GCFIRE . . . . .	IV-7-B-32
9. Routine GCTDET . . . . .	IV-7-B-38
10. Routine GCKILL . . . . .	IV-7-B-44
11. Routine GCKLL1 . . . . .	IV-7-B-45
12. Routine GPKILL . . . . .	IV-7-B-47
13. Routine GBKEEP . . . . .	IV-7-B-48
14. Routine MURFAC . . . . .	IV-7-B-51
15. Routine GLEG . . . . .	IV-7-B-53
16. Routine GCLAST . . . . .	IV-7-B-55
17. Routine GUPDAT . . . . .	IV-7-B-61

## CONTENTS (Continued)

	<u>Page</u>
Appendix C. Ground Combat Model Output Descriptions . . . . .	IV-7-C-1
1. Introduction . . . . .	IV-7-C-1
2. Unit Geometry and Target Acquisition . . . . .	IV-7-C-1
3. Engagement Results . . . . .	IV-7-C-7
4. Mortar Scheduling . . . . .	IV-7-C-9
Appendix D. Source Listings for Period Processor Ground Combat Model . . . . .	IV-7-D-1
Appendix E. References . . . . .	IV-7-E-1
Chapter 8. AREA FIRE MODEL . . . . .	IV-8-1
1. Military Activity Represented . . . . .	IV-8-1
2. Model Design . . . . .	IV-8-1
3. Submodel Specifications . . . . .	IV-8-4
4. History Tape Output . . . . .	IV-8-21
Appendix A. Area Fire Model Input Requirements . . . . .	IV-8-A-1
1. Introduction . . . . .	IV-8-A-1
2. Data Matrices for the Area Fire Model . . . . .	IV-8-A-3
3. Weapon/Munition Delivery Characteristics Data Card . . . . .	IV-8-A-9
4. Weapons/Munition Projectile Description . . . . .	IV-8-A-12
5. Weapon/Munition Lethal Areas for Personnel . . . . .	IV-8-A-14
6. Lethal Area for Enemy Force Equipment . . . . .	IV-8-A-16
7. Personnel Posture . . . . .	IV-8-A-18
8. Personnel Protection by Organic Equipment . . . . .	IV-8-A-21
9. Area Fire Data Deck Structure . . . . .	IV-8-A-24
Appendix B. Area Fire Model Program Description . . . . .	IV-8-B-1
1. Introduction . . . . .	IV-8-B-1
2. Routine FIREDT . . . . .	IV-8-B-1
3. Routine AREAFR . . . . .	IV-8-B-8
4. Routine AKILL . . . . .	IV-8-B-23
5. Routine OASSES . . . . .	IV-8-B-24
Appendix C. Area Fire Model Output Descriptions . . . . .	IV-8-C-1
1. Introduction . . . . .	IV-8-C-1
2. Area Fire Model Output . . . . .	IV-8-C-1
Appendix D. Source Listings for Period Processor Area Fire Model . . . . .	IV-8-D-1

## CONTENTS (Continued)

	<u>Page</u>
Appendix E. References . . . . .	IV-8-E-1
Chapter 9. TACTICAL FIRE DIRECTION SYSTEM MODEL . . . . .	IV-9-1
1. Military Activity Represented . . . . .	IV-9-1
2. Model Design . . . . .	IV-9-1
3. Submodel Specifications . . . . .	IV-9-4
Appendix A. Tactical Fire Direction System Input Requirements . . . . .	IV-9-A-1
1. Introduction . . . . .	IV-9-A-1
2. Tacfire Data Base . . . . .	IV-9-A-1
3. Data Required . . . . .	IV-9-A-3
4. Preparation of the TACFIRE Data Base for Card Input. . . . .	IV-9-A-17
5. Summary of Data Base Preparation . . . . .	IV-9-A-25
6. TACFIRE Constant Data Deck Structure . . . . .	IV-9-A-25
Appendix B. TACFIRE Model Program Description . . . . .	IV-9-B-1
1. Introduction . . . . .	IV-9-B-1
2. Routine OVER9 . . . . .	IV-9-B-1
3. Routine AFTFC1 . . . . .	IV-9-B-1
4. Routine AFTFC2 . . . . .	IV-9-B-17
Appendix C. TACFIRE Model Output Descriptions . . . . .	IV-9-C-1
1. Introduction . . . . .	IV-9-C-1
2. TACFIRE Mission Scheduling . . . . .	IV-9-C-1
Appendix D. Source Listings for Period Processor TACFIRE Model . . . . .	IV-9-D-1
Appendix E. References . . . . .	IV-9-E-1
Chapter 10. AIR GROUND ENGAGEMENT MODEL . . . . .	IV-10-1
1. Military Activity Represented . . . . .	IV-10-1
2. Model Design . . . . .	IV-10-4
3. Submodel Specifications . . . . .	IV-10-8
4. History Tape Output . . . . .	IV-10-27
Appendix A. Air Ground Engagement Model Input Requirements . . . . .	IV-10-A-1
1. Introduction . . . . .	IV-10-A-1
2. Card Format Relationships . . . . .	IV-10-A-1

## CONTENTS (Continued)

	<u>Page</u>
3. Data Input by Mission Type (2601) . . . . .	IV-10-A-2
4. Data Input by Aircraft Type (2602) . . . . .	IV-10-A-13
5. Data Input by Weapon Type (2603) . . . . .	IV-10-A-24
6. Results for Various Air Strike Situations (2701) . . .	IV-10-A-35
7. Air Ground Engagement Constant Data Deck Structure . .	IV-10-A-42
 Appendix B. Air Ground Engagement Model Program Descriptions .	IV-10-B-1
1. Introduction . . . . .	IV-10-B-1
2. Routine AIRGND . . . . .	IV-10-B-1
3. Routine BTF . . . . .	IV-10-B-2
4. Routine ATB . . . . .	IV-10-B-9
5. Routine AVAIL . . . . .	IV-10-B-15
6. Routine PREPTM . . . . .	IV-10-B-16
7. Routine TIMES . . . . .	IV-10-B-19
8. Routine SELECT . . . . .	IV-10-B-24
9. Routine ENRATA . . . . .	IV-10-B-25
10. Routine TORA . . . . .	IV-10-B-36
11. Routine BTA . . . . .	IV-10-B-44
12. Routine SEG5 . . . . .	IV-10-B-49
 Appendix C. Air Ground Engagement Model Output Descriptions .	IV-10-C-1
1. Introduction . . . . .	IV-10-C-1
2. Air Mission Request Routine ATB . . . . .	IV-10-C-1
3. En Route Attrition - Routine ENRATA . . . . .	IV-10-C-9
4. Engagement Results - Routine TORA . . . . .	IV-10-C-17
5. Safe Point Update - Routine BTF . . . . .	IV-10-C-20
6. Mission End - Routine BTA . . . . .	IV-10-C-21
 Appendix D. Source Listings for Period Processor Air Ground Engagement Model . . . . .	IV-10-D-1
 Appendix E. References . . . . .	IV-10-E-1
 Chapter 11. SUPPRESSION MODEL . . . . .	IV-11-1
1. Introduction . . . . .	IV-11-1
2. Suppression of Movement . . . . .	IV-11-1
3. Suppression of Indirect Fire . . . . .	IV-11-1
4. Suppression of Air Defense Fire . . . . .	IV-11-1
5. Interfaces with Other Models . . . . .	IV-11-2
 Appendix A. Suppression Model Input Requirements . . . . .	IV-11-A-1
1. Introduction . . . . .	IV-11-A-1
2. Assembling the Data Base . . . . .	IV-11-A-2

## CONTENTS (Continued)

	<u>Page</u>
3. Suppression Constant Data Deck Structure . . . . .	IV-11-A-5
Appendix B. Suppression Model Program Descriptions . . . . .	IV-11-B-1
1. Introduction . . . . .	IV-11-B-1
2. Routine SUPRES . . . . .	IV-11-B-1
Appendix C. Suppression Model Output Description . . . . .	IV-11-C-1
1. Introduction . . . . .	IV-11-C-1
2. Routine SUPRES . . . . .	IV-11-C-1
Appendix D. Source Listings for Period Processor Suppression Model . . . . .	IV-11-D-1
Chapter 12. NUCLEAR ASSESSMENT MODEL . . . . .	IV-12-1
1. Military Activity Represented . . . . .	IV-12-1
2. Model Design . . . . .	IV-12-2
3. Submodel Specifications . . . . .	IV-12-14
Appendix A. Nuclear Assessment Model Input Requirements . . .	IV-12-A-1
1. Introduction . . . . .	IV-12-A-1
2. Data Requirements . . . . .	IV-12-A-1
3. Weapon/Munition Characteristics . . . . .	IV-12-A-8
4. Weapon Effects . . . . .	IV-12-A-11
5. Equipment Vulnerability . . . . .	IV-12-A-15
6. Barrier/Facility Vulnerability . . . . .	IV-12-A-17
7. Personnel Postures . . . . .	IV-12-A-17
8. Radiation Barriers . . . . .	IV-12-A-23
9. Nuclear Model Constant Data Deck Structure . . . . .	IV-12-A-25
Appendix B. Nuclear Assessment Model Program Descriptions . .	IV-12-B-1
1. Introduction . . . . .	IV-12-B-1
2. Routine NAM . . . . .	IV-12-B-1
3. Routine NAM4 . . . . .	IV-12-B-2
4. Routines NAM1 and NAM2 . . . . .	IV-12-B-2
5. Routine NFIRE . . . . .	IV-12-B-4
6. Routine NFIRDT . . . . .	IV-12-B-6
7. Routine BURST . . . . .	IV-12-B-8
8. Routine NDRD . . . . .	IV-12-B-10
9. Routine SUPSCH . . . . .	IV-12-B-12
10. Routine ASUNIT . . . . .	IV-12-B-14
11. Routine HEMOVN . . . . .	IV-12-B-18



CONTENTS (Continued)

	<u>Page</u>
12. Routine UPRMPT . . . . .	IV-12-B-20
13. Routine NASPER . . . . .	IV-12-B-22
14. Routine NASEOH . . . . .	IV-12-B-25
15. Routine FNDLNK . . . . .	IV-12-B-29
16. Routine FNDLOS . . . . .	IV-12-B-30
17. Routine NPEPL . . . . .	IV-12-B-33
18. Routine STUPRO . . . . .	IV-12-B-34
19. Routine CFUN . . . . .	IV-12-B-36
20. Routine BANCIR . . . . .	IV-12-B-38
21. Routine NBARAS . . . . .	IV-12-B-40
22. Routine BARUPE . . . . .	IV-12-B-43
23. Routine BARNUM . . . . .	IV-12-B-44
24. Routine NCUNIT . . . . .	IV-12-B-47
25. Routine BAMUNT . . . . .	IV-12-B-49
 Appendix C. Nuclear Assessment Model Output Descriptions . .	 IV-12-C-1
1. Introduction . . . . .	IV-12-C-1
2. Routine NFIRE . . . . .	IV-12-C-1
3. Routine NBARAS . . . . .	IV-12-C-6
4. Routine NCUNIT . . . . .	IV-12-C-8
5. Routine BAMUNT . . . . .	IV-12-C-9
 Appendix D. Source Listings for Period Processor Nuclear Assessment Model . . . . .	  IV-12-D-1
 Appendix E. References . . . . .	 IV-12-E-1

CONTENTS (Continued)

PART 4

	<u>Page</u>
TITLE PAGE . . . . .	1
CONTENTS . . . . .	.iii
FIGURES . . . . .	x
SECTION IV PERIOD PROCESSOR (Continued)	
Chapter 13. MOVEMENT MODEL . . . . .	IV-13-1
1. Military Unit Movement . . . . .	IV-13-1
2. Model Design . . . . .	IV-13-1
3. Submodel Specifications . . . . .	IV-13-9
4. History Tape Output . . . . .	IV-13-21
Appendix A. Movement Model Input Requirements . . . . .	IV-13-A-1
1. Introduction . . . . .	IV-13-A-1
2. Assembling the Data Base . . . . .	IV-13-A-4
3. Travel Mode and Mobility Category . . . . .	IV-13-A-18
4. UTD by Mobility Category . . . . .	IV-13-A-19
5. Mobility Classification . . . . .	IV-13-A-22
6. UTD Class Exclusion List . . . . .	IV-13-A-24
7. Unit Road Movement Rate . . . . .	IV-13-A-24
8. Vehicular Road Movement Rates . . . . .	IV-13-A-28
9. Unit Cross Country Movement Rate . . . . .	IV-13-A-30
10. Vehicular Cross Country Movement Rates . . . . .	IV-13-A-33
11. Ground Movement Fuel Consumption Rate . . . . .	IV-13-A-35
12. Aircraft Fuel Consumption Rates . . . . .	IV-13-A 37
13. Breach Force Decision Tables . . . . .	IV-13-A-40
14. Movement Model Constant Data Deck Structure . . . . .	IV-13-A-40
Appendix B. Movement Model Program Descriptions . . . . .	IV-13-B-1
1. Introduction . . . . .	IV-13-B-1
2. Routine MOVESR . . . . .	IV-13-B-1
3. Routine MOVEDT . . . . .	IV-13-B-7
4. Routine OBSRCH . . . . .	IV-13-B-10
5. Routine MOVE . . . . .	IV-13-B-14
6. Routine OVER13 . . . . .	IV-13-B-17
Appendix C. Movement Model Output Descriptions . . . . .	IV-13-C-1
1. Introduction . . . . .	IV-13-C-1
2. Routine MOVF . . . . .	IV-13-C-1

CONTENTS (Continued)

	<u>Page</u>
3. Routine MOVEDT . . . . .	IV-13-C-1
4. Routine MOVESR . . . . .	IV-13-C-5
5. Routine OBSRCH . . . . .	IV-13-C-7
Appendix D. Source Listings for Period Processor Movement Model . . . . .	IV-13-D-1
Chapter 14. ENGINEER MODEL . . . . .	IV-14-1
1. Military Activity Represented . . . . .	IV-14-1
2. Model Design . . . . .	IV-14-2
3. Submodel Specifications . . . . .	IV-14-8
Appendix A. Engineer Model Input Requirements . . . . .	IV-14-A-1
1. Introduction . . . . .	IV-14-A-1
2. Data Base for Barriers and Facilities . . . . .	IV-14-A-2
3. Data Base for Engineer Task . . . . .	IV-14-A-11
4. Engineer Constant Data Deck Structure . . . . .	IV-14-A-29
Appendix B. Engineer Model Program Descriptions . . . . .	IV-14-B-1
1. Introduction . . . . .	IV-14-B-1
2. Routine ENGR . . . . .	IV-14-B-2
3. Routine EPRIOR . . . . .	IV-14-B-12
4. Routine PRORTY . . . . .	IV-14-B-15
5. Routine EFEASI . . . . .	IV-14-B-19
6. Routine FESBIL . . . . .	IV-14-B-23
7. Routine ELELEA . . . . .	IV-14-B-28
8. Routine EUPDAT . . . . .	IV-14-B-32
9. Routine BUIDRC . . . . .	IV-14-B-38
10. Routine CREA16 . . . . .	IV-14-B-38
11. Routines EQPT and EQPTUP . . . . .	IV-14-B-40
12. Routine GEOM . . . . .	IV-14-B-40
13. Routine IRECN1 . . . . .	IV-14-B-40
14. Routine MPRALC . . . . .	IV-14-B-40
15. Routine MPRUPD . . . . .	IV-14-B-44
16. Routines STUNIT and SET37 . . . . .	IV-14-B-44
17. Routine UNTORG . . . . .	IV-14-B-44
18. Routine CRTQD . . . . .	IV-14-B-44
19. Routine NUCSCH . . . . .	IV-14-B-45
20. Routine ENUCLE . . . . .	IV-14-B-48
Appendix C. Engineer Model Output Descriptions . . . . .	IV-14-C-1
1. Introduction . . . . .	IV-14-C-1
2. Additional Print Statements . . . . .	IV-14-C-9
Appendix D. Source Listings for Period Processor Engineer Model . . . . .	IV-14-D-1

## CONTENTS (Continued)

	<u>Page</u>
Chapter 15. AIRMOBILE MODEL . . . . .	IV-15-1
1. Military Activities Represented . . . . .	IV-15-1
2. Design of the Airmobile Model . . . . .	IV-15-8
3. Submodel Specifications . . . . .	IV-15-9
Appendix A. Airmobile Model Input Requirements . . . . .	IV-15-A-1
1. Introduction . . . . .	IV-15-A-1
2. Data Input by Other Load Programs . . . . .	IV-15-A-1
3. Second Priority Equipment Items . . . . .	IV-15-A-1
4. Mix Table . . . . .	IV-15-A-2
5. Forward Area Rearm and Refuel (FARR) Area Definitions . . . . .	IV-15-A-6
6. Refuel Device Definition . . . . .	IV-15-A-6
7. Suppression Mission Data (GTAA) . . . . .	IV-15-A-9
8. Acquisition Data . . . . .	IV-15-A-1
9. Aircraft Data . . . . .	IV-15-A-15
10. Air Defense Weapon Data . . . . .	IV-15-A-17
11. Airmobile Constant Data Deck Structure . . . . .	IV-15-A-30
Appendix B. Airmobile Model Program Descriptions . . . . .	IV-15-B-1
1. Introduction . . . . .	IV-15-B-1
2. Routine AMBLEX . . . . .	IV-15-B-1
3. Routine ACCPT . . . . .	IV-15-B-1
4. Routine ACCPT1 . . . . .	IV-15-B-4
5. Routine ACCPT2 . . . . .	IV-15-B-14
6. Routine ACCPT3 . . . . .	IV-15-B-17
7. Routine ASULT . . . . .	IV-15-B-19
8. Routine ASULT1 . . . . .	IV-15-B-20
9. Routine ASULT2 . . . . .	IV-15-B-25
10. Routine ASULT3 . . . . .	IV-15-B-30
11. Routine ASULT4 . . . . .	IV-15-B-3
12. Common Block FRRA . . . . .	IV-15-B-39
13. Routine RREXEC . . . . .	IV-15-B-41
14. Routine RRFRRRA . . . . .	IV-15-B-42
15. Routine RRQUE . . . . .	IV-15-B-50
16. Routine RRFLD . . . . .	IV-15-B-55
17. Routine REFUEL . . . . .	IV-15-B-63
18. Routine RATIM . . . . .	IV-15-B-67
19. Routine RRTURN . . . . .	IV-15-B-68
20. Routine RLEAS . . . . .	IV-15-B-70
21. Routine RLEAS1 . . . . .	IV-15-B-73
22. Routine RLEAS2 . . . . .	IV-15-B-78
23. Routine RLEAS3 . . . . .	IV-15-B-81
24. Routine RLEAS4 . . . . .	IV-15-B-86
25. Routine PENRAT . . . . .	IV-15-B-88
26. Routine SETTYP . . . . .	IV-15-B-90

## CONTENTS (Continued)

	<u>Page</u>
27. Routine CIRCLE . . . . .	IV-15-B-92
28. Routine MORS . . . . .	IV-15-B-94
29. Routine CIRCLE . . . . .	IV-15-B-94
30. Routine GASRCH . . . . .	IV-15-B-95
31. Routine ATTDVR . . . . .	IV-15-B-99
32. Routine MES . . . . .	IV-15-B-102
33. Routine SIS . . . . .	IV-15-B-112
34. Routine GTAA . . . . .	IV-15-B-120
35. Routine LOSSES . . . . .	IV-15-B-132
 Appendix C. Airmobile Model Output Descriptions . . . . .	 IV-15-C-1
1. Introduction . . . . .	IV-15-C-1
2. Routine ACCPT1 . . . . .	IV-15-C-1
3. Routine ACCPT2 . . . . .	IV-15-C-4
4. Routine ACCPT3 . . . . .	IV-15-C-4
5. Routine ASULT1 . . . . .	IV-15-C-4
6. Routine ASULT2 . . . . .	IV-15-C-4
7. Routine ASULT3 . . . . .	IV-15-C-8
8. Routine ASULT4 . . . . .	IV-15-C-8
9. Routine RLEAS1 . . . . .	IV-15-C-8
10. Routine RLEAS2 . . . . .	IV-15-C-11
11. Routine RLEAS3 . . . . .	IV-15-C-11
12. Routine RLEAS4 . . . . .	IV-15-C-14
13. Rearm Refuel Submodel . . . . .	IV-15-C-14
14. Routine SIS . . . . .	IV-15-C-19
15. Routine GTAA . . . . .	IV-15-C-23
16. Routine LOSSES . . . . .	IV-15-C-27
17. Routine MES . . . . .	IV-15-C-29
 Appendix D. Source Listing for Period Processor	
Airmobile Model . . . . .	IV-15-D-1
 Appendix E. References . . . . .	 IV-15-E-1
 Chapter 16. COMBAT SERVICE SUPPORT MODEL . . . . .	 IV-16-1
1. Military Actions Represented . . . . .	IV-16-1
2. Model Design . . . . .	IV-16-7
3. Submodel Specifications . . . . .	IV-16-9
4. History Tape Output . . . . .	IV-16-23
 Appendix A. Combat Service Support Model Input Requirements . .	 IV-16-A-1
1. Introduction . . . . .	IV-16-A-1
2. Elapsed Time to Handle Transports and Class I	
Consumption Rates . . . . .	IV-16-A-3
3. Items by Supply Class . . . . .	IV-16-A-5
4. Weights and Volumes Carried by Various Transports . . .	IV-16-A-8
5. Resupply of Class VII and Personnel . . . . .	IV-16-A-10
6. Combat Service Support Constant Data Deck Structure . .	IV-16-A-13

CONTENTS (Continued)

	<u>Page</u>
Appendix B. Combat Service Support Model Program Descriptions	IV-16-B-1
1. Introduction . . . . .	IV-16-B-1
2. Routine SERSUP . . . . .	IV-16-B-1
3. Routine TRNTIM . . . . .	IV-16-B-24
4. Routine EVTFIL . . . . .	IV-16-B-25
5. Routine SUPFIL . . . . .	IV-16-B-28
6. Routine BOFILE . . . . .	IV-16-B-32
7. Routine FETCH . . . . .	IV-16-B-33
8. Routine MISOUT . . . . .	IV-16-B-35
9. Routine RESLVE . . . . .	IV-16-B-37
10. Routine CSSANL . . . . .	IV-16-B-37
Appendix C. Combat Service Support Model Output Descriptions	IV-16-C-1
Appendix D. Source Listings for Period Processor Combat Service Support Model . . . . .	IV-16-D-1
Chapter 17. MAJOR SPECIAL PURPOSE PROGRAMS . . . . .	IV-17-1
1. Introduction . . . . .	IV-17-1
2. Initialization . . . . .	IV-17-1
3. Termination . . . . .	IV-17-1
4. Restart . . . . .	IV-17-1
5. Common Tables . . . . .	IV-17-1
6. Food Consumption . . . . .	IV-17-1
Appendix A. Input Requirements for the Major Special Purpose Programs . . . . .	IV-17-A-1
Appendix B. Major Special Purpose Program Descriptions . . . . .	IV-17-B-1
1. Introduction . . . . .	IV-17-B-1
2. Routine INITIAL . . . . .	IV-17-B-1
3. Routine LOCTAP . . . . .	IV-17-B-5
4. Routine COMMDP . . . . .	IV-17-B-5
5. Routine XXIT . . . . .	IV-17-B-5
6. Routine EXIT1 . . . . .	IV-17-B-7
7. Routine SET41 . . . . .	IV-17-B-9
8. Routine EAT . . . . .	IV-17-B-10
Appendix C. Output Descriptions of Major Special Purpose Routines . . . . .	IV-17-C-1
1. Introduction . . . . .	IV-17-C-1
2. Routine INITIAL . . . . .	IV-17-C-1
3. Routine EXIT1 . . . . .	IV-17-C-3
4. Routine COMMDP . . . . .	IV-17-C-4
5. Routine UTILDP . . . . .	IV-17-C-4

<u>CONTENTS (Continued)</u>	<u>Page</u>
Appendix D. Source Listings for Period Processor Major Special Purpose Routines . . . . .	IV-17-D-1
Chapter 18. DIVWAG SCENARIO LANGUAGE INTERFACE ROUTINES . . . . .	IV-18-1
1. General Description . . . . .	IV-18-1
2. Routine DOSR . . . . .	IV-18-1
3. Routine DBSR . . . . .	IV-18-1
4. Routine TRUTH . . . . .	IV-18-1
5. Routine BSUIDL . . . . .	IV-18-2
Appendix A. Input Requirements for DIVWAG Scenario Language Interface Routines . . . . .	IV-18-A-1
Appendix B. DIVWAG Scenario Language Interface Program Descriptions . . . . .	IV-18-B-1
1. Introduction . . . . .	IV-18-B-1
2. Routine DSLINT . . . . .	IV-18-B-1
3. Routine DOSR . . . . .	IV-18-B-1
4. Routine DBSR . . . . .	IV-18-B-4
5. Routine BSUIDL . . . . .	IV-18-B-5
6. Routine TRUTH . . . . .	IV-18-B-7
Appendix C. Output Descriptions for DIVWAG Scenario Language Interface Routines . . . . .	IV-18-C-1
Appendix D. Source Listings for Period Processor DIVWAG Scenario Language Interface . . . . .	IV-18-D-1
Chapter 19. UTILITY ROUTINES . . . . .	IV-19-1
1. Introduction . . . . .	IV-19-1
2. Organization . . . . .	IV-19-1
Appendix A. Input Requirements for Utility Routines . . . . .	IV-19-A-1
Appendix B. Program Descriptions for Utility Routines . . . . .	IV-19-B-1
1. Introduction . . . . .	IV-19-B-1
2. Routine SETCLK . . . . .	IV-19-B-1
3. Routine MINGET . . . . .	IV-19-B-1
4. Routine REDCLK . . . . .	IV-19-B-1
5. Routine ARCTAN . . . . .	IV-19-B-1
6. Routine CHORD . . . . .	IV-19-B-2
7. Routine DISTPL . . . . .	IV-19-B-3
8. Routine DSTPL1 . . . . .	IV-19-B-4
9. Routine INTSPT . . . . .	IV-19-B-4
10. Routine PONTLN . . . . .	IV-19-B-7

CONTENTS (Continued)

	<u>Page</u>
11. Routine CELLST . . . . .	IV-19-B-9
12. Routine LOS . . . . .	IV-19-B-13
13. Routine SEARCH . . . . .	IV-19-B-16
14. Routine INTPOL . . . . .	IV-19-B-19
15. Routine RANGEF . . . . .	IV-19-B-21
16. Routines RANKA and RANK . . . . .	IV-19-B-21
17. Routine SCALE . . . . .	IV-19-B-22
18. Routine DNORM . . . . .	IV-19-B-24
19. Routine ENORM . . . . .	IV-19-B-24
20. Routine FNORM . . . . .	IV-19-B-26
21. Routine RANDU . . . . .	IV-19-B-26
22. Routine RAND1 . . . . .	IV-19-B-27
23. Routine IUIDF . . . . .	IV-19-B-27
24. Routine ADDUNT . . . . .	IV-19-B-29
25. Routine EOH2OT . . . . .	IV-19-B-31
26. Routine PUTOUT . . . . .	IV-19-B-32
27. Routine SCORE . . . . .	IV-19-B-35
28. Routine BUFOUT . . . . .	IV-19-B-37
29. Routine BUFFIN . . . . .	IV-19-B-37
30. Routine KPICK . . . . .	IV-19-B-39
31. Routine BLKDAT . . . . .	IV-19-B-39
32. Routine GET . . . . .	IV-19-B-39
33. Routine PUT . . . . .	IV-19-B-41
 Appendix C. Output Descriptions for Utility Routines . . . . .	 IV-19-C-1
 Appendix D. Source Listings for Period Processor	
Utility Routines . . . . .	IV-19-D-1



## CONTENTS (Continued)

### PART 5

	<u>Page</u>
TITLE PAGE . . . . .	i
CONTENTS . . . . .	iii
FIGURES . . . . .	xiii
SECTION V. PERIOD OUTPUT PROCESSOR . . . . .	V-1
Chapter 1. GENERAL DESCRIPTION OF THE PERIOD OUTPUT PROCESSOR . .	V-1-1
1. Overview . . . . .	V-1-1
2. General Reports . . . . .	V-1-1
Chapter 2. EXECUTIVE CONTROL AND UTILITY ROUTINES . . . . .	V-2-1
1. Introduction . . . . .	V-2-1
2. REPCTL . . . . .	V-2-1
3. INITAL . . . . .	V-2-1
4. HEADER . . . . .	V-2-1
Appendix A. Executive Control and Utility Program Descriptions	V-2-A-1
1. Introduction . . . . .	V-2-A-1
2. Routine REPCTL . . . . .	V-2-A-1
3. Routine INITAL . . . . .	V-2-A-4
4. Routine HEADER . . . . .	V-2-A-5
Appendix B. Source Listings for Period Output Processor	
Executive Control and Utility Routines . . . . .	V-2-B-1
Chapter 3. STATUS AND ACTIVITY REPORTS . . . . .	V-3-1
1. Introduction . . . . .	V-3-1
2. Status File Updating System Flow . . . . .	V-3-1
3. Unit Status Reporting System Flow . . . . .	V-3-1
Appendix A. Input Requirements for Status and Activity Reports	V-3-A-1
Appendix B. Status and Activity Reports Program Descriptions .	V-3-B-1
1. Introduction . . . . .	V-3-B-1
2. Routine UPEOH . . . . .	V-3-B-1
3. Routine DNCHAI . . . . .	V-3-B-8
4. Routine ZERO . . . . .	V-3-B-9
5. Routine GETUSE . . . . .	V-3-B-9
6. Routine USTAT1 . . . . .	V-3-B-9

## CONTENTS (Continued)

	<u>Page</u>
7. Routine PUTDET . . . . .	V-3-B-21
8. Routine EOHLOC . . . . .	V-3-B-32
9. Routine EOHPER . . . . .	V-3-B-37
10. Routine UACTR . . . . .	V-3-B-42
11. Routine UACTY . . . . .	V-3-B-47
12. Routine OVFL . . . . .	V-3-B-48
13. Routine GET31 . . . . .	V-3-B-48
14. Routine SUM . . . . .	V-3-B-49
15. Routine GETSUM . . . . .	V-3-B-53
Appendix C. Status and Activity Reports Output Descriptions. .	V-3-C-1
Appendix D. Source Listings for Period Output Processor Status and Activity Reports . . . . .	V-3-D-1
Chapter 4. PLANNING REPORT . . . . .	V-4-1
1. Introduction . . . . .	V-4-1
2. System Flow . . . . .	V-4-1
Appendix A. Input Requirements for Planning Report . . . . .	V-4-A-1
Appendix B. Planning Report Program Descriptions . . . . .	V-4-B-1
1. Introduction . . . . .	V-4-B-1
2. Routine UIDS01 . . . . .	V-4-B-1
3. Routine PLREPT . . . . .	V-4-B-3
4. Routine PLOVFL . . . . .	V-4-B-4
Appendix C. Planning Report Output Description . . . . .	V-4-C-1
Appendix D. Source Listing for Period Output Processor Planning Report . . . . .	V-4-D-1
Chapter 5. INTELLIGENCE REPORT . . . . .	V-5-1
1. Introduction . . . . .	V-5-1
2. System Flow . . . . .	V-5-1
Appendix A. Input Requirements for Intelligence Report . . . .	V-5-A-1
Appendix B. Intelligence Report Program Description . . . . .	V-5-B-1
1. Introduction . . . . .	V-5-B-1
2. Routine INTLRP . . . . .	V-5-B-1
Appendix C. Intelligence Report Output Description . . . . .	V-5-C-1

CONTENTS (Continued)

	<u>Page</u>
Appendix D. Source Listings for Period Output Processor Intelligence Report . . . . .	V-5-D-1
Chapter 6. BARRIER REPORT . . . . .	V-6-1
1. Introduction . . . . .	V-6-1
2. System Flow . . . . .	V-6-1
Appendix A. Input Requirements for Barrier Report . . . . .	V-6-A-1
Appendix B. Barrier Report Program Descriptions . . . . .	V-6-B-1
1. Introduction . . . . .	V-6-B-1
2. Routine ENGREL . . . . .	V-6-B-1
3. Routine GET2 . . . . .	V-6-B-4
Appendix C. Barrier Report Output Description . . . . .	V-6-C-1
Appendix D. Source Listings for Period Output Processor Barrier Report . . . . .	V-6-D-1
SECTION VI. ANALYSIS OUTPUT PROCESSOR . . . . .	VI-1
Chapter 1. GENERAL DESCRIPTION OF ANALYSIS OUTPUT PROCESSOR . . .	VI-1-1
1. Introduction . . . . .	VI-1-1
2. Overview . . . . .	VI-1-1
3. Logical Flow . . . . .	VI-1-1
Appendix A. History Tape Formats . . . . .	VI-1-A-1
1. Introduction . . . . .	VI-1-A-1
2. History Tape Records . . . . .	VI-1-A-1
Chapter 2. TAPE PREPROCESSORS . . . . .	VI-2-1
1. Introduction . . . . .	VI-2-1
2. Routine PREP . . . . .	VI-2-1
3. Routine PTAPE . . . . .	VI-2-1
4. Routine UXR . . . . .	VI-2-1
Appendix A. Input Requirements for Tape Preprocessors . . . .	VI-2-A-1
1. Introduction . . . . .	VI-2-A-1
2. Routine PREP . . . . .	VI-2-A-1
3. Routine UXR . . . . .	VI-2-A-1
4. Routine PTAPE . . . . .	VI-2-A-1

## CONTENTS (Continued)

	<u>Page</u>
Appendix B. History Tape Preprocessors Program Descriptions .	VI-2-B-1
1. Introduction . . . . .	VI-2-B-1
2. Routine PREP . . . . .	VI-2-B-1
3. Routine UXR . . . . .	VI-2-B-4
4. Routine PTAPE . . . . .	VI-2-B-4
5. Routine INITAL . . . . .	VI-2-B-7
6. Routine UIDCD . . . . .	VI-2-B-7
7. Routine UTDCD . . . . .	VI-2-B-13
8. Routine CLSCD . . . . .	VI-2-B-13
9. Routine TIMCD . . . . .	VI-2-B-15
10. Routine DEVCD . . . . .	VI-2-B-17
11. Routine ENDCD . . . . .	VI-2-B-21
12. Routine UIDGEN . . . . .	VI-2-B-25
13. Routine SQUEZ . . . . .	VI-2-B-25
14. Routine DRSPLT . . . . .	VI-2-B-26
15. Routine READR . . . . .	VI-2-B-27
16. Routine SAVDSI . . . . .	VI-2-B-29
17. Routine WRTDOR . . . . .	VI-2-B-30
Appendix C. Output Descriptions of Tape Preprocessor Routines.	VI-2-C-1
1. Introduction . . . . .	VI-2-C-1
2. Output Descriptions . . . . .	VI-2-C-1
3. Routine PREP . . . . .	VI-2-C-1
4. Routine UXR . . . . .	VI-2-C-1
5. Routine PTAPE . . . . .	VI-2-C-4
Appendix D. Source Listings for Analysis Output Processor Tape Preprocessors . . . . .	VI-2-D-1
Chapter 3. HISTORY TAPE LISTING ROUTINES . . . . .	VI-3-1
Appendix A. Input Requirements for History Tape Listing Program . . . . .	VI-3-A-1
1. Introduction . . . . .	VI-3-A-1
2. Routine PHIST . . . . .	VI-3-A-1
Appendix B. History Tape Listing Program Descriptions . . . .	VI-3-B-1
1. Introduction . . . . .	VI-3-B-1
2. Routine PHIST . . . . .	VI-3-B-1
3. History Record Print Routines . . . . .	VI-3-B-3
Appendix C. Output Descriptions of History Tape Listing Routines . . . . .	VI-3-C-1
1. Introduction . . . . .	VI-3-C-1

## CONTENTS (Continued)

	<u>Page</u>
2. Area Fire Assessment Record . . . . .	VI-3-C-1
3. Area Fire Event Record . . . . .	VI-3-C-1
4. Air Ground Mission Request Record . . . . .	VI-3-C-4
5. Air Ground En Route Attrition Record . . . . .	VI-3-C-4
6. Air Ground Target Engagement Record . . . . .	VI-3-C-7
7. Air Ground Target Assessment Record . . . . .	VI-3-C-7
8. Ground Combat Attacker-Defender Pair Record . . . . .	VI-3-C-10
9. Ground Combat Battle Unit Record . . . . .	VI-3-C-12
10. Combat Service Support Events Records . . . . .	VI-3-C-12
11. Movement Event Records . . . . .	VI-3-C-20
 Appendix D. Source Listings for Analysis Output Processor	
History Tape Routines . . . . .	VI-3-D-1
 Chapter 4. ANALYSIS MATRIX FORMULATION . . . . .	VI-4-1
1. Introduction . . . . .	VI-4-1
2. Routine ANCARD . . . . .	VI-4-1
3. Routines AFM and AGM . . . . .	VI-4-1
4. Routine GCMOD . . . . .	VI-4-2
 Appendix A. Input Requirements for Analysis Matrix	
Formulation Programs . . . . .	VI-4-A-1
1. Introduction . . . . .	VI-4-A-1
2. Routine ANCARD . . . . .	VI-4-A-1
3. Routine AFM . . . . .	VI-4-A-4
4. Routine AGM . . . . .	VI-4-A-4
5. Routine GCMOD . . . . .	VI-4-A-6
 Appendix B. Analysis Matrix Formulation Program Descriptions .	VI-4-B-1
1. Introduction . . . . .	VI-4-B-1
2. Routine ANCARD . . . . .	VI-4-B-1
3. Routine GCMOD . . . . .	VI-4-B-12
4. Routine MATOUT . . . . .	VI-4-B-20
5. Routine MPRCES . . . . .	VI-4-B-21
6. Routine AGM . . . . .	VI-4-B-33
7. Routine AFM . . . . .	VI-4-B-39
8. Routine IOMAT . . . . .	VI-4-B-45
9. Routine TAPEND . . . . .	VI-4-B-49
10. Routine ERROR . . . . .	VI-4-B-50
11. Routine NINELD . . . . .	VI-4-B-50
12. Routine SETUP . . . . .	VI-4-B-50
 Appendix C. Output Descriptions of Analysis Matrix Formulation	
Routines . . . . .	VI-4-C-1
1. Introduction . . . . .	VI-4-C-1

CONTENTS (Continued)

	<u>Page</u>
2. Routine ANCARD . . . . .	VI-4-C-1
3. Routine AFM . . . . .	VI-4-C-1
4. Routine AGM . . . . .	VI-4-C-1
5. Routine GCMOD . . . . .	VI-4-C-4
Appendix D. Source Listings for Analysis Output Processor Analysis Matrix Formulation . . . . .	VI-4-D-1
Chapter 5. ANALYSIS ROUTINES . . . . .	VI-5-1
1. Introduction . . . . .	VI-5-1
2. Routine MIXMRG . . . . .	VI-5-1
3. Routine MTXUP . . . . .	VI-5-1
4. Routine ANALYS . . . . .	VI-5-1
Appendix A. Input Requirements for Analysis Programs . . . . .	VI-5-A-1
1. Introduction . . . . .	VI-5-A-1
2. Routine MTXMRG . . . . .	VI-5-A-1
3. Routine MTXUP . . . . .	VI-5-A-1
4. Routine ANALYS . . . . .	VI-5-A-1
Appendix B. Analysis Routines Program Descriptions . . . . .	VI-5-B-1
1. Introduction . . . . .	VI-5-B-1
2. Routine MTXMRG . . . . .	VI-5-B-1
3. Routine MTXUP . . . . .	VI-5-B-4
4. Routine ANALYS . . . . .	VI-5-B-8
5. Routine CALCH . . . . .	VI-5-B-9
6. Routine PAIR . . . . .	VI-5-B-14
7. Routine FRDMAN . . . . .	VI-5-B-21
8. Routine DSESRT . . . . .	VI-5-B-21
9. Routine FRGFIT . . . . .	VI-5-B-23
10. Routine ANOVA . . . . .	VI-5-B-25
Appendix C. Output Descriptions of Analysis Routines . . . . .	VI-5-C-1
1. Introduction . . . . .	VI-5-C-1
2. Routine MTXMRG . . . . .	VI-5-C-1
3. Routine MTXUP . . . . .	VI-5-C-1
4. Routine ANALYS . . . . .	VI-5-C-4
Appendix D. Source Listings for Analysis Output Processor Analysis Routines . . . . .	VI-5-C-1

## CONTENTS (Continued)

	<u>Page</u>
SECTION VII DIVWAG SYSTEM UTILITY PACKAGES . . . . .	VII-1
Chapter 1. GENERAL DESCRIPTIONS OF DIVWAG SYSTEM UTILITY PACKAGES	VII-1-1
1. Introduction . . . . .	VII-1-1
2. Utility Load and Dump Routines . . . . .	VII-1-1
3. DIVWAG Input/Output Routines . . . . .	VII-1-1
4. Other Utility Routines . . . . .	VII-1-1
Chapter 2. DIVWAG DATA FILES . . . . .	VII-2-1
Appendix A. Data File Formats . . . . .	VII-2-A-1
1. Unit Status File . . . . .	VII-2-A-1
2. Barrier-Facility File . . . . .	VII-2-A-5
3. Terrain File . . . . .	VII-2-A-8
4. Weather File . . . . .	VII-2-A-8
5. Logical File 5 . . . . .	VII-2-A-9
6. Secondary Equipment File . . . . .	VII-2-A-9
7. Airmobile Data File . . . . .	VII-2-A-9
8. Suppression Data File . . . . .	VII-2-A-11
9. UTD Mobility Category File . . . . .	VII-2-A-12
10. Logical File 10 . . . . .	VII-2-A-13
11. Combat Service Support Data File . . . . .	VII-2-A-13
12. Automatic Event File . . . . .	VII-2-A-17
13. Elevation File . . . . .	VII-2-A-17
14. Movement Data . . . . .	VII-2-A-17
15. Fuel Consumption Rates . . . . .	VII-2-A-19
16. Period Processor Scratch File . . . . .	VII-2-A-20
17. Engineer Rate and Task Characteristics . . . . .	VII-2-A-23
18. Engineer Task File . . . . .	VII-2-A-26
19. Unit Movement Rate Tables . . . . .	VII-2-A-26
20. Sensor Data . . . . .	VII-2-A-27
21. Battle Records . . . . .	VII-2-A-42
22. Engineer Task Site File . . . . .	VII-2-A-43
23. Unit Composition Data . . . . .	VII-2-A-43
24. Airmobile Rearm/Refuel Point Data . . . . .	VII-2-A-43
25. TACFIRE Data . . . . .	VII-2-A-45
26. Air Ground Engagement Data File . . . . .	VII-2-A-49
27. Engagement Result Tables File . . . . .	VII-2-A-50
28. Unit Structure and Item Distribution Data . . . . .	VII-2-A-51
29. Munition Lethal Areas and Delivery Data . . . . .	VII-2-A-52
30. Personnel Posture and Protection Data and Nuclear Data	VII-2-A-53
31. Supply Status File . . . . .	VII-2-A-57
32. Data File 32 . . . . .	VII-2-A-58
33. Data File 33 . . . . .	VII-2-A-58
34. Data File 34 . . . . .	VII-2-A-58
35. Data File 35 . . . . .	VII-2-A-58
36. Common Storage File . . . . .	VII-2-A-58

## CONTENTS (Continued)

	<u>Page</u>
37. Engineer Equipment File . . . . .	VII-2-A-63
38. Data File 38 . . . . .	VII-2-A-63
39. Ground Combat Model Data . . . . .	VII-2-A-63
40. Data File 40 . . . . .	VII-2-A-66
41. Battalion Target Information File . . . . .	VII-2-A-66
42. Brigade/Regiment Target Information File . . . . .	VII-2-A-68
43. Division Target Information File . . . . .	VII-2-A-68
44. Redundant Target File . . . . .	VII-2-A-68
45. Camera Detections File . . . . .	VII-2-A-68
46. Data File 46 . . . . .	VII-2-A-69
47. Data File 47 . . . . .	VII-2-A-69
48. Score Board File . . . . .	VII-2-A-69
49. Data File 49 . . . . .	VII-2-A-69
50. Authorized Strength Data . . . . .	VII-2-A-69
51. UTD Directory . . . . .	VII-2-A-70
52. Basic Unit Descriptions . . . . .	VII-2-A-70
53. Unit Type Composition . . . . .	VII-2-A-70
54. Data File 54 . . . . .	VII-2-A-70
55. DSL Orders Table . . . . .	VII-2-A-71
 Chapter 3. DIVWAG INPUT/OUTPUT PACKAGE . . . . .	 VII-3-1
1. Introduction . . . . .	VII-3-1
2. DIVWAG Data File Logical Layout . . . . .	VII-3-1
 Appendix A. Input Requirements for Input/Output Package . . .	 VII-3-A-1
 Appendix B. DIVWAG Input/Output Package Program Descriptions .	 VII-3-B-1
1. Introduction . . . . .	VII-3-B-1
2. Routine CREATE . . . . .	VII-3-B-1
3. Routine REMOVE . . . . .	VII-3-B-5
4. Routine ADDRCD . . . . .	VII-3-B-8
5. Routine ADDWRD . . . . .	VII-3-B-12
6. Routine SUBRCD . . . . .	VII-3-B-17
7. Routine SUBWRD . . . . .	VII-3-B-20
8. Routine GETFLE . . . . .	VII-3-B-21
9. Routine PUTFLE . . . . .	VII-3-B-24
10. Routine FILE . . . . .	VII-3-B-25
11. Routine GETRCD . . . . .	VII-3-B-27
12. Routine PTRCD . . . . .	VII-3-B-27
13. Routine RECORD . . . . .	VII-3-B-28
14. Routine GETWRD . . . . .	VII-3-B-31
15. Routine PUTWRD . . . . .	VII-3-B-32
16. Routine WORD . . . . .	VII-3-B-32
17. Routine GETPUT . . . . .	VII-3-B-35
18. Routine TRNSMT . . . . .	VII-3-B-36
19. Routine FSL . . . . .	VII-3-B-46
20. Routine SHFTDN . . . . .	VII-3-B-48
21. Routine ILLEGL . . . . .	VII-3-B-51



CONTENTS (Continued)

	<u>Page</u>
22. Routine NCOMP . . . . .	VII-3-B-52
23. Routine OPENMS . . . . .	VII-3-B-52
24. Routine MASSIO . . . . .	VII-3-B-52
Appendix C. Output Descriptions for Input/Output Package . . .	VII-3-C-1
Appendix D. Source Listings for DIVWAG System Utility Packages, Input/Output Package . . . . .	VII-3-D-1
Chapter 4. UTILITY LOAD AND DUMP PROGRAMS . . . . .	VII-4-1
Appendix A. Input Requirements for Utility Load and Dump Programs . . . . .	VII-4-A-1
1. UTILDP . . . . .	VII-4-A-1
2. UTILLD . . . . .	VII-4-A-1
Appendix B. DIVWAG Input and Output Utility Program Descriptions . . . . .	VII-4-B-1
1. Introduction . . . . .	VII-4-B-1
2. Routine IOTERN . . . . .	VII-4-B-1
3. Routine IOWETH . . . . .	VII-4-B-4
4. Routine UTILLD . . . . .	VII-4-B-5
5. Routine UTILDP . . . . .	VII-4-B-10
Appendix C. Output Descriptions of Utility Load and Dump Programs . . . . .	VII-4-C-1
1. Introduction . . . . .	VII-4-C-1
2. Routine UTILLD . . . . .	VII-4-C-1
3. Routine UTILDP . . . . .	VII-4-C-1
Appendix D. Source Listings for DIVWAG System Utility Packages, Utility Load and Dump Programs . . . . .	VII-4-D-1
Chapter 5. OTHER UTILITY ROUTINES . . . . .	VII-5-1
1. Introduction . . . . .	VII-5-1
2. Character Packing . . . . .	VII-5-1
3. Six-Bit Pack . . . . .	VII-5-1
4. Integer/Character Conversion . . . . .	VII-5-1
5. Air Ground Engagement Data . . . . .	VII-5-1
Appendix A. Input Requirements for Other Utility Routines . .	VII-5-A-1
Appendix B. Program Descriptions for Other Utility Routines .	VII-5-B-1

CONTENTS (Continued)

	<u>Page</u>
1. Introduction . . . . .	VII-5-B-1
2. Routine MVCHAR . . . . .	VII-5-B-1
3. Routine KHAR . . . . .	VII-5-B-3
4. Routine PACK28 . . . . .	VII-5-B-3
5. Routine INTCHR . . . . .	VII-5-B-5
6. Routine SNATCH . . . . .	VII-5-B-7
Appendix C. Output Descriptions for Other Utility Routines . .	VII-5-C-1
Appendix D. Source Listings for DIVWAG System Utility Packages, Other Utility Routines . . . . .	VII-5-D-1

CONTENTS (Continued)

PART 6

	<u>Page</u>
TITLE PAGE . . . . .	1
CONTENTS . . . . .	iii
SECTION VIII. DIVWAG DIAGNOSTIC HANDBOOK . . . . .	VIII-1
Chapter 1. INTRODUCTION . . . . .	VIII-1-1
1. Purpose . . . . .	VIII-1-1
2. Scope . . . . .	VIII-1-1
3. Organization . . . . .	VIII-1-1
Chapter 2. CONSTANT DATA INPUT PROCESSOR ERROR CONDITIONS . . . . .	VIII-2-1
1. Introduction . . . . .	VIII-2-1
2. Executive Control . . . . .	VIII-2-1
3. Organization and Equipment . . . . .	VIII-2-1
4. Task Organization . . . . .	VIII-2-5
5. Environmental Load Routines . . . . .	VIII-2-11
6. Unit Geometry . . . . .	VIII-2-12
7. Intelligence and Control . . . . .	VIII-2-13
8. Ground Combat . . . . .	VIII-2-16
9. Area Fire . . . . .	VIII-2-17
10. TACFIRE . . . . .	VIII-2-20
11. Air Ground Engagement . . . . .	VIII-2-21
12. Suppression . . . . .	VIII-2-23
13. Nuclear Assessment . . . . .	VIII-2-24
14. Movement . . . . .	VIII-2-28
15. Engineer . . . . .	VIII-2-31
16. Airmobile . . . . .	VIII-2-35
17. Combat Service Support . . . . .	VIII-2-35
Chapter 3. ORDERS INPUT PROCESSOR ERROR CONDITIONS . . . . .	VIII-3-1
1. Introduction . . . . .	VIII-3-1
2. DSL Compiler . . . . .	VIII-3-1
3. Operating Instructions . . . . .	VIII-3-11
Chapter 4. PERIOD PROCESSOR ERROR CONDITIONS . . . . .	VIII-4-1
1. Introduction . . . . .	VIII-4-1
2. Executive Control . . . . .	VIII-4-1
3. Unit Representation . . . . .	VIII-4-7
4. Environmental Representation . . . . .	VIII-4-11
5. Battlefield and Unit Geometry . . . . .	VIII-4-11

## CONTENTS (Continued)

	<u>Page</u>
6. Intelligence and Control . . . . .	VIII-4-12
7. Ground Combat . . . . .	VIII-4-14
8. Area Fire . . . . .	VIII-4-14
9. TACFIRE . . . . .	VIII-4-14
10. Air Ground Engagement . . . . .	VIII-4-15
11. Suppression . . . . .	VIII-4-17
12. Nuclear Assessment . . . . .	VIII-4-17
13. Movement . . . . .	VIII-4-18
14. Engineer . . . . .	VIII-4-19
15. Airmobile . . . . .	VIII-4-20
16. Combat Service Support . . . . .	VIII-4-22
17. Special Purpose Routines . . . . .	VIII-4-24
18. DSL Interface . . . . .	VIII-4-24
19. Utility Routines . . . . .	VIII-4-26
 Chapter 5. PERIOD OUTPUT PROCESSOR ERROR CONDITIONS . . . . .	 VIII-5-1
1. Introduction . . . . .	VIII-5-1
2. Executive Control . . . . .	VIII-5-1
3. Status of Activity Report . . . . .	VIII-5-1
4. Planning Report . . . . .	VIII-5-2
5. Intelligence Report . . . . .	VIII-5-2
6. Barrier Report . . . . .	VIII-5-3
 Chapter 6. ANALYSIS OUTPUT PROCESSOR ERROR CONDITIONS . . . . .	 VIII-6-1
1. Introduction . . . . .	VIII-6-1
2. Tape Preprocessors . . . . .	VIII-6-1
3. History Tape Listing Routine . . . . .	VIII-6-3
4. Matrix Formulation Routines . . . . .	VIII-6-3
5. Statistical Analysis Routines . . . . .	VIII-6-6
 Chapter 7. DIVWAG INPUT/OUTPUT PACKAGE ERROR CONDITIONS . . . . .	 VIII-7-1
1. Introduction . . . . .	VIII-7-1
2. Error Diagnostics . . . . .	VIII-7-1
3. Error Messages . . . . .	VIII-7-3
 Chapter 8. UTILITY LOAD AND DUMP ERROR CONDITIONS . . . . .	 VIII-8-1
1. Introduction . . . . .	VIII-8-1
2. Utility Load . . . . .	VIII-8-1
3. Utility Dump . . . . .	VIII-8-1

## FIGURES

### PART 1

<u>Number</u>		<u>Page</u>
I-3-1	DIVWAG System Data Flow	I-3-4
I-3-2	Processor Input and Output	I-3-5
II-2-A-1	Executive Job Control Deck Assembly	II-2-A-2
II-2-A-2	Executive Control Constant Data Control Deck Assembly	II-2-A-3
II-2-B-1	Routine PREDIV	II-2-B-2
II-3-B-1	Routine LDTOE	II-3-B-2
II-3-B-2	Routine SRCH	II-3-B-15
II-3-B-3	Routine DMPTOE	II-3-B-17
II-3-B-4	Routine SORT	II-3-B-21
II-3-C-1	TOE Type Printouts	II-3-C-1
II-3-C-2	80-Column Card Image	II-3-C-2
II-3-C-3	Secondary Equipment for Blue	II-3-C-3
II-3-C-4	Unit Type Designator Breakdown Table	II-3-C-5
II-3-C-5	Unit Type Designator Directory	II-3-C-6
II-3-C-6	Authorized Personnel and Equipment Printout	II-3-C-7
II-4-B-1	Routine ECHLON	II-4-B-3
II-4-B-2	Routine RECOG1	II-4-B-9
II-4-B-3	Routine COMMCK	II-4-B-17
II-4-B-4	Routine COMPRS	II-4-B-19
II-4-B-5	Routine CORDIN	II-4-B-21
II-4-B-6	Routine SUPORT	II-4-B-23
II-4-B-7	Routine SEQUEN	II-4-B-26
II-4-B-8	Routine TOEPUT	II-4-B-33
II-4-B-9	Routine UPDATE	II-4-B-42
II-4-B-10	Routine CRAT31	II-4-B-51
II-4-B-11	Routine RELATR	II-4-B-52
II-4-B-12	Routine SUPPLY	II-4-B-56
II-4-B-13	Routine TALLY	II-4-B-61
II-4-C-1	Unit Type Designator Directory, Task Organization	II-4-C-2
II-4-C-2	Card Image, Task Organization	II-4-C-3
II-4-C-3	RECOGN1 Table, Task Organization	II-4-C-4
II-4-C-4	Supply Status Table, Task Organization	II-4-C-5
II-4-C-5	Task Organization Unit Resolution List	II-4-C-6
II-4-C-6	Unit Type Designator Summary, Task Organization	II-4-C-8
II-4-C-7	Task Organization Units Created	II-4-C-9
II-5-B-1	Record Layout of Terrain Elevation Tape	II-5-B-2
II-5-B-2	Routine ELEVLD	II-5-B-3
II-5-B-3	Routine TERNLD	II-5-B-8
II-5-B-4	Routine WETHLD	II-5-B-13
II-5-C-1	Weather File Printout	II-5-C-2

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
II-6-B-1	Routine LOAD28	II-6-B-4
II-6-B-2	Routine DUMP28	II-6-B-12
II-6-C-1	Data File 28 80-Column Card Image	II-6-C-2
II-6-C-2	Unit Type Designator Record Number Directory	II-6-C-3
II-6-C-3	Unit Description	II-6-C-4
II-7-B-1	Routine INCSLD	II-7-B-2
II-7-B-2	Routine SET20	II-7-B-13
II-7-C-1	Printouts of Intelligence and Control Constant Data Input	II-7-C-1
II-7-C-2	INC 80-Column Card Image	II-7-C-3
II-7-C-3	Constant Data on Sensors	II-7-C-4
II-7-C-4	Unattended Ground Sensors for Blue Force	II-7-C-5
II-7-C-5	Equipment Item Code for Target Category	II-7-C-6
II-7-C-6	Standing Operating Procedure for Reporting Hostile Forces	II-7-C-7
II-7-C-7	Flow Delay Times	II-7-C-8
II-7-C-8	Processing and Decision Delay Times	II-7-C-9
II-7-C-9	Unit Size Based on Equipment and Personnel Counts	II-7-C-10
II-7-C-10	Target Inferences	II-7-C-12
II-8-B-1	Routine GCMLD	II-8-B-2
II-8-B-2	Routine DUMP39	II-8-B-9
II-8-C-1	Ground Combat Printouts for Constant Data Input	II-8-C-1
II-8-C-2	Ground Combat 80-Column Card Image	II-8-C-2
II-8-C-3	Line of Sight	II-8-C-3
II-8-C-4	Target and Pinpoint Probabilities	II-8-C-5
II-8-C-5	Weapon System Specifications	II-8-C-6
II-8-C-6	Weapon/Target Kill Probabilities	II-8-C-7
II-8-C-7	Sensor Distribution and Performance	II-8-C-9
II-9-B-1	Routine AFMLD	II-8-B-2
II-9-B-2	Routine LOAD29	II-8-B-6
II-9-B-3	Routine LOAD30	II-8-B-14
II-9-B-4	Routine DUMP29	II-9-B-22
II-9-B-5	Routine DUMP30	II-9-B-24
II-9-C-1	Area Fire Constant Data Printouts	II-9-C-1
II-9-C-2	Data File 29 80-Column Card Image	II-9-C-2
II-9-C-3	Weapons and Munitions Characteristics and Performance	II-9-C-3
II-9-C-4	Weapons and Munitions Delivery Parameters	II-9-C-4
II-9-C-5	Weapons and Munitions Description	II-9-C-6
II-9-C-6	Data File 30 80-Column Card Image	II-9-C-7
II-9-C-7	Personnel Data	II-9-C-8
II-10-B-1	Routine TACLD	II-10-B-2
II-10-B-2	Routine LOAD25	II-10-B-4
II-10-B-3	Routine OSORT	II-10-B-15

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
II-10-B-4	Routine DUMP25	II-10-B-18
II-10-C-1	TACFIRE Constant Data Printouts	II-10-C-1
II-10-C-2	Data File 25 80-Column Card Image	II-10-C-2
II-10-C-3	Priority and Method of Attack	II-10-C-3
II-10-C-4	Summary of Target Priorities	II-10-C-4
II-11-B-1	Routine AIRLD	II-11-B-4
II-11-C-1	Air Ground Engagement Constant Data Printouts	II-11-C-2
II-11-C-2	Data File 26 80-Column Card Image	II-11-C-3
II-11-C-3	Aircraft and Munitions Minimum/Maximum Requirements Mission Type	II-11-C-4
II-11-C-4	Aircraft Preparation Delay Times	II-11-C-5
II-11-C-5	Aircraft Engagement Parameters	II-11-C-6
II-11-C-6	Minimum Aircraft Abort Data	II-11-C-7
II-11-C-7	Aircraft Landing Times	II-11-C-9
II-11-C-8	Aircraft Flight Speeds	II-11-C-10
II-11-C-9	Aircraft Availability Times for New Mission	II-11-C-11
II-11-C-10	Aircraft Degradation for Maintenance	II-11-C-12
II-11-C-11	Aircraft Vulnerable Areas for Kills by Weapon Type	II-11-C-13
II-11-C-12	Air Defense Characteristics	II-11-C-14
II-11-C-13	Air Defense Weapons Effectiveness	II-11-C-15
II-11-C-14	Air Defense Weapon Degradation from Terrain and Vegetation	II-11-C-16
II-11-C-15	Engagement Results for Air Strikes	II-11-C-18
II-12-B-1	Routine SPRSLD	II-12-B-2
II-12-C-1	Suppression Model Data Card Listing	II-12-C-2
II-13-B-1	Routine NUCLD	II-13-B-2
II-13-B-2	Routine FLD30	II-13-B-6
II-13-B-3	Routine CKARY	II-13-B-10
II-13-C-1	Nuclear Assessment Model Constant Data Input Printouts	II-13-C-1
II-13-C-2	Nuclear Assessment 80-Column Card Image	II-13-C-2
II-13-C-3	Weapon Combination Names	II-13-C-4
II-13-C-4	Weapon Statistics	II-13-C-4
II-13-C-5	Fuze Table Entries	II-13-C-5
II-13-C-6	Yield Table	II-13-C-6
II-13-C-7	EOH Links to Damage Radii	II-13-C-7
II-13-C-8	EOH Data	II-13-C-8
II-13-C-9	Yield Codes and Barrier Data	II-13-C-9
II-13-C-10	Unprotected Personnel Distribution	II-13-C-10
II-13-C-11	Special Factors	II-13-C-10
II-14-B-1	Routine MOVELD	II-14-B-2
II-14-C-1	Movement Model Constant Data Input Printouts	II-14-C-1
II-14-C-2	Movement Model 80-Column Card Image	II-14-C-3

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
II-14-C-3	Mobility Category and Travel Mode	II-14-C-4
II-14-C-4	Mobility Class and Exclusion Table	II-14-C-5
II-14-C-5	Unit Road and Cross Country Movement Rates	II-14-C-6
II-14-C-6	Equipment Road and Cross Country Movement Rates	II-14-C-7
II-14-C-7	Ground and Air Fuel Consumption Rates	II-14-C-8
II-15-B-1	Routine ENGLD	II-15-B-2
II-15-B-2	Routine BARLD	II-15-B-10
II-15-C-1	Barrier Segment Location and Status	II-15-C-2
II-15-C-2	Engineer 80-Column Card Image	II-15-C-3
II-15-C-3	Engineering Task Parameters	II-15-C-4
II-15-C-4	Environmental and Troop Rates and Rate Modifiers	II-15-C-5
II-15-C-5	Equipment Requirements and Rate Modifiers	II-15-C-6
II-15-C-6	Minefield Forcing Casualties	II-15-C-7
II-16-B-1	Routine FIL7LD	II-16-B-2
II-16-C-1	Sample of 80-Column Card Image, Airmobile Model	II-16-C-2
II-16-C-2	Sample Data Listing, Airmobile Model	II-16-C-3
II-17-B-1	Routine CSSLD	II-17-B-2
II-17-B-2	Routine LOAD11	II-17-B-5
II-17-B-3	Routine DUMP11	II-17-B-11
II-17-C-1	Combat Service Support Constant Data Printouts	II-17-C-1
II-17-C-2	Combat Service Support 80-Column Card Image	II-17-C-2
II-17-C-3	Elapsed Time to Handle Vehicles	II-17-C-3
II-17-C-4	Weight and Volume of Consumables and Capacity of Vehicles	II-17-C-5
II-17-C-5	Equipment Item Versus Class of Supply	II-17-C-6
II-17-C-6	Transport Preference for Hauling Consumables	II-17-C-7
II-17-C-7	Blue Major End Items and Personnel Available Daily for Resupply	II-17-C-8
III-2-1	Unit Battle Table Description	III-2-2
III-2-2	Numerical Order Codes	III-2-3
III-2-3	Conditional Type Codes	III-2-4
III-2-4	Standard Order Column Descriptions	III-2-5
III-2-5	Airmobile Order Column Descriptions	III-2-6
III-2-6	Engineer Order Column Descriptions	III-2-7
III-2-7	Conditional Column Descriptions	III-2-8
III-2-8	DSL Compiler Logical Flow	III-2-9
III-2-9	DSL Logical Flow	III-2-11
III-2-10	Routine DSLCMP Logical Flow	III-2-12
III-2-A-1	Travel Mode Mnemonic Description	III-2-A-21
III-2-A-2	DSL Data Deck Structure	III-2-A-32
III-2-A-3	Delay and Range Settings for Mohawk Type Mission	III-2-A-38
III-2-A-4	DSL Orders	III-2-A-39
III-2-A-5	DSL Order Modifiers	III-2-A-42



# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
III-2-B-1	Routine DSLCMP	III-2-B-7
III-2-B-2	Routine DSLINT	III-2-B-9
III-2-B-3	Routine RDSTMT	III-2-B-11
III-2-B-4	Routine BATORD	III-2-B-14
III-2-B-5	Routine UNTORD	III-2-B-17
III-2-B-6	Routine COND	III-2-B-22
III-2-B-7	Routine AIRMOB	III-2-B-27
III-2-B-8	Routine ENGR	III-2-B-29
III-2-B-9	Routine FIRE	III-2-B-31
III-2-B-10	Routine FLY	III-2-B-32
III-2-B-11	Routine MOVE	III-2-B-34
III-2-B-12	Routine STAY	III-2-B-36
III-2-B-13	Routine TRANS	III-2-B-37
III-2-B-14	Routine KRAKMD	III-2-B-39
III-2-B-15	Routine STOW	III-2-B-48
III-2-B-16	Routine PASS2	III-2-B-50
III-2-B-17	Routine PAGE	III-2-B-52
III-2-B-18	Routine TIME	III-2-B-54
III-2-B-19	Routine CHCKID	III-2-B-57
III-2-B-20	Routine MATCH	III-2-B-60
III-2-B-21	Routine FINDN	III-2-B-62
III-2-C-1	Routine DSLINT Printed Sample Output	III-2-C-2
III-2-C-2	Routine RDSTMT Printed Sample Output	III-2-C-3
III-2-C-3	Routine PAGE Printed Sample Output	III-2-C-3
III-2-C-4	Routine DUMPF Printed Sample Output	III-2-C-5
III-3-A-1	Moving Target Indicator Radar Card Format	III-3-A-3
III-3-A-2	Countermortar/Counterbattery Radar Card Format	III-3-A-5
III-3-A-3	Air Defense Radar Card Format	III-3-A-7
III-3-A-4	Unattended Ground Sensor Identification Card Format	III-3-A-9
III-3-A-5	Unattended Ground Sensor Field Composition Card Format	III-3-A-11
III-3-A-6	Unattended Ground Sensor Location Card Format	III-3-A-12
III-3-A-7	Geometric Figures	III-3-A-13
III-3-A-8	Period Card Format	III-3-A-15
III-3-A-9	Intelligence Information Card Format	III-3-A-16
III-3-A-10	Estimated Equipment Card Format	III-3-A-19
III-3-A-11	Helicopter and Artillery Penetration Limits Card Format	III-3-A-20
III-3-A-12	Blue Sorties Card Format	III-3-A-22
III-3-A-13	Red Sorties Card Format	III-3-A-23
III-3-A-14	Data Deck Arrangement for Operating Instructions	III-3-A-25
III-3-B-1	Routine DATAST	III-3-B-2
III-3-B-2	Routine VINKLD	III-3-B-6
III-3-B-3	Routine LDASR	III-3-B-14
III-3-B-4	Routine INTLIN	III-3-B-20
III-3-C-1	Sample Printed Output by the Operating Instructions Process	III-3-C-2

# FIGURES (Continued)

## PART 2

<u>Number</u>		<u>Page</u>
IV-1-1	Basic Event Sequencing Logic	IV-1-3
IV-1-2	Event Scheduling Within DIVWAG System	IV-1-4
IV-1-3	DIVWAG System Models	IV-1-6
IV-1-4	Period Processor Program Groups	IV-1-8
IV-2-1	Executive Structure	IV-2-2
IV-2-2	Event Table	IV-2-3
IV-2-A-1	Period Processor Control Card	IV-2-A-2
IV-2-B-1	Routine MAIN	IV-2-B-2
IV-2-B-2	Routine MINUET	IV-2-B-3
IV-2-B-3	Routine ESR	IV-2-B-26
IV-2-B-4	Routine ARCNRL	IV-2-B-31
IV-2-B-5	Routine FORCRD	IV-2-B-39
IV-2-B-6	Routine ENGORD	IV-2-B-41
IV-2-B-7	Routine EVTSET	IV-2-B-43
IV-2-B-8	Routine PUTEVT	IV-2-B-44
IV-2-B-9	Routine GETEVT	IV-2-B-46
IV-2-B-10	Routine INRUPT	IV-2-B-48
IV-2-C-1	Routine MINUET Printed Output Sample	IV-2-C-2
IV-3-1	Brigade Organization at Battalion Level Resolution	IV-3-2
IV-3-A-1	TOE Flow	IV-3-A-3
IV-3-A-2	Unit Organization Chart	IV-3-A-8
IV-3-A-3	Example of Item Code List of Equipment for DIVWAG	IV-3-A-9
IV-3-A-4	Senior/Subordinate Units	IV-3-A-14
IV-3-A-5	Secondary Equipment List	IV-3-A-16
IV-3-A-6	Table of Organization and Equipment	IV-3-A-18
IV-3-A-7	Bulk-loaded Expendable Supplies	IV-3-A-21
IV-3-A-8	LDTOE Data Deck Structure	IV-3-A-22
IV-3-A-9	TOE Data Deck Structure	IV-3-A-24
IV-3-A-10	Examples of Use of UID	IV-3-A-27
IV-3-A-11	Sample Bookkeeping Form for Relating Units to UTD and UID	IV-3-A-30
IV-3-A-12	Diagram of Supply Doctrine for Classes I, III, and IV and Personnel	IV-3-A-32
IV-3-A-13	Elementary Organizational Example	IV-3-A-35
IV-3-A-14	Elementary Organizational Input Example	IV-3-A-35
IV-3-A-15	Example of Task Organizational Input	IV-3-A-36
IV-3-A-16	Supply Point Data	IV-3-A-41
IV-3-A-17	Task Organization Data Deck	IV-3-A-43
IV-3-B-1	Routine TRFR	IV-3-B-2
IV-3-B-2	Routine ASSIGN	IV-3-B-5
IV-3-B-3	Routine ASSUME	IV-3-B-7
IV-3-B-4	Routine DETACH	IV-3-B-11
IV-3-B-5	Routine DTCH31	IV-3-B-18
IV-3-B-6	Routine JOIN	IV-3-B-23

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
IV-3-B-7	Routine BIAS	IV-3-B-28
IV-3-B-8	Routine JOIN31	IV-3-B-29
IV-4-1	Trafficability Index	IV-4-5
IV-4-2	Use of Terrain Description with DIVWAG System	IV-4-7
IV-4-3	Dominant Mask Feature	IV-4-11
IV-4-4	Dominant Masking Function	IV-4-12
IV-4-A-1	Example Overlay Depicting Grid Cells for Total Geographical Area	IV-4-A-2
IV-4-A-2	Terrain Parameter Card	IV-4-A-3
IV-4-A-3	Terrain File Load Card	IV-4-A-5
IV-4-A-4	Soil/Trafficability Index Definitions	IV-4-A-9
IV-4-A-5	Weather Zone Table Card Format	IV-4-A-12
IV-4-A-6	Weather Report Card Format	IV-4-A-14
IV-4-A-7	Visibility Index	IV-4-A-15
IV-4-A-8	Temperature Gradient Factor	IV-4-A-17
IV-4-A-9	Weather Data Deck Structure	IV-4-A-19
IV-4-B-1	Routine UPMASK	IV-4-B-2
IV-4-B-2	Routine ELEVAT	IV-4-B-4
IV-4-B-3	Routine WETTHR	IV-4-B-7
IV-4-C-1	Routine UPMASK Printed Output Sample from Environmental Representation	IV-4-C-2
IV-5-1	Battlefield Orientation	IV-5-2
IV-5-2	Unit Boundaries	IV-5-4
IV-5-3	Unit Geometry	IV-5-6
IV-5-A-1	Completed Unit Geometry Planning Matrix	IV-5-A-2
IV-5-A-2	UTDs with Common Dimensions and Resources Distribution	IV-5-A-5
IV-5-A-3	Comparison of UTDs	IV-5-A-6
IV-5-A-4	Equipments Issued to UTDs with Common Dimensions	IV-5-A-9
IV-5-A-5	Bands and Unit Dimensions	IV-5-A-11
IV-5-A-6	Unit Distribution of Personnel and Equipment	IV-5-A-13
IV-5-A-7	Area Fire Card Format Listings	IV-5-A-15
IV-5-A-8	Unit Geometry Data Deck Structure	IV-5-A-15
IV-5-B-1	Routine ZONE	IV-5-B-2
IV-5-B-2	Routine AFLAG	IV-5-B-5
IV-5-B-3	Routine UPDVUL	IV-5-B-8
IV-5-C-1	Routine ZONE Printed Output Sample from Battlefield and Unit Geometry	IV-5-C-2
IV-5-C-2	Routine UPDVUL Printed Output Sample from Battlefield and Unit Geometry	IV-5-C-2
IV-6-1	Intelligence and Control Activities Simulated	IV-6-2
IV-6-2	Intelligence and Control Model Macroflow	IV-6-9
IV-6-3	Collection Submodel Macroflow	IV-6-13
IV-6-4	Targets Subunits	IV-6-14

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
IV-6-5	Intelligence and Control Model Line of Sight RBAR Parameters	IV-6-16
IV-6-6	Typical Intelligence and Control Model Line of Sight Curves	IV-6-17
IV-6-7	Air/Reconnaissance/Surveillance Macroflow	IV-6-18
IV-6-8	RECON1 Route and Area Reconnaissance	IV-6-21
IV-6-9	Search Sector Geometry for RECON1	IV-6-23
IV-6-10	Recognition and Identification versus Resolved Target Elements or Signal-to-Noise	IV-6-32
IV-6-11	Mohawk SLAR Range Gating	IV-6-36
IV-6-12	Search Geometry for Ground MTI Radar	IV-6-48
IV-6-13	Entry of UGS Field	IV-6-51
IV-6-14	Radar Probability of Collection Curves (MTI Ground Radar)	IV-6-55
IV-6-15	Unit Location Error for MTI Ground-Based Radar	IV-6-58
IV-6-16	Macroflow of UCOLLF	IV-6-60
IV-6-17	Geometry of Countermortar/Counterbattery Artillery Fire Acquisition	IV-6-61
IV-6-18	Principal Items in a File Record	IV-6-71
IV-6-19	Threshold Matrix for Battalion to Pass Information or Intelligence to Other Echelons	IV-6-77
IV-6-20	Threshold Matrix for Brigade to Pass Information or Intelligence to Other Echelons	IV-6-78
IV-6-21	Threshold Matrix for Division to Pass Information or Intelligence to Other Echelons	IV-6-79
IV-6-22	Fire Support Decision Table	IV-6-84
IV-6-23	Damage Levels as a Function of Separation of Center of Impact and Sensor Location	IV-6-88
IV-6-24	Sensor Status Evaluation	IV-6-89
IV-6-25	UGS Field Assessment Geometry	IV-6-91
IV-6-A-1	General Radar Data	IV-6-A-3
IV-6-A-2	Sensor Category	IV-6-A-4
IV-6-A-3	General Radar Data, Continued	IV-6-A-8
IV-6-A-4	Radar Range	IV-6-A-10
IV-6-A-5	Target Type Code for Radar	IV-6-A-11
IV-6-A-6	General Countermortar/Counterbattery Radar Data	IV-6-A-14
IV-6-A-7	General Countermortar/Counterbattery Radar Data, Continued	IV-6-A-16
IV-6-A-8	Ground Based Radar Range Brackets for Sensor Movement	IV-6-A-18
IV-6-A-9	Sensor Constant Data Input Format	IV-6-A-20
IV-6-A-10	Airborne Visual Observers	IV-6-A-21
IV-6-A-11	Intelligence and Control Target Codes and Indexes	IV-6-A-26
IV-6-A-12	Airborne SLAR/MTI	IV-6-A-27
IV-6-A-13	Aerial Cameras	IV-6-A-31

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
IV-6-A-14	Air Reconnaissance Model Sensor Combination Load Input Format	IV-6-A-33
IV-6-A-15	LOH Decision Control Matrix	IV-6-A-34
IV-6-A-16	LOH or Army Aviation Fixed Wing Reconnaissance Vehicle TOE	IV-6-A-36
IV-6-A-17	Reconnaissance Equipment and Amount-TOE	IV-6-A-37
IV-6-A-18	Delay Times for Each Unit Type Designation	IV-6-A-39
IV-6-A-19	Normal Interval Between Senior and Subordinate Units	IV-6-A-42
IV-6-A-20	Target Inferences from Counts and Intervals	IV-6-A-44
IV-6-A-21	Units and Abbreviations	IV-6-A-45
IV-6-A-22	Target Inferences from Counts and Intervals, Continued	IV-6-A-49
IV-6-A-23	Thresholds of Information Dissemination	IV-6-A-52
IV-6-A-24	Headquarters Originating and Receiving Intelligence Messages	IV-6-A-53
IV-6-A-25	Abbreviations for Type Units	IV-6-A-55
IV-6-A-26	Activity Code and Description	IV-6-A-56
IV-6-A-27	Unattended Ground Sensors Input Data Example	IV-6-A-58
IV-6-A-28	Unattended Ground Sensors	IV-6-A-59
IV-6-A-29	Vehicles and Weapons Codes	IV-6-A-61
IV-6-A-30	Target Code Categories	IV-6-A-63
IV-6-A-31	Intelligence and Control Model Data Deck Listing	IV-6-A-64
IV-6-A-32	Intelligence and Control Data Deck Structure	IV-6-A-65
IV-6-B-1	Routine INCEXC	IV-6-B-2
IV-6-B-2	Routine CCOLLM	IV-6-B-7
IV-6-B-3	Routine COLLECT	IV-6-B-18
IV-6-B-4	Routine CCOLLF	IV-6-B-31
IV-6-B-5	Routine WORKIT	IV-6-B-39
IV-6-B-6	Routine PLNINT	IV-6-B-44
IV-6-B-7	Routine CIRINT	IV-6-B-47
IV-6-B-8	Routine CKINT	IV-6-B-49
IV-6-B-9	Routine IQANGC	IV-6-B-51
IV-6-B-10	Routine CPDT	IV-6-B-54
IV-6-B-11	Routine PROCES	IV-6-B-59
IV-6-B-12	Routine UTSR	IV-6-B-72
IV-6-B-13	Routine SUTR	IV-6-B-75
IV-6-B-14	Routine TAR	IV-6-B-78
IV-6-B-15	Routine ARESPO	IV-6-B-80
IV-6-B-16	Routine DECIDT	IV-6-B-82
IV-6-B-17	Routine DECIDE	IV-6-B-85
IV-6-B-18	Routine MOVSEN	IV-6-B-99
IV-6-B-19	Routine SASSMT	IV-6-B-105
IV-6-B-20	Routine SENSRH	IV-6-B-112
IV-6-B-21	Routine UPSENH	IV-6-B-114
IV-6-B-22	Routine RCMAIN	IV-6-B-116

FIGURES (Continued)

<u>Number</u>		<u>Page</u>
IV-6-B-23	Routine NUMTGT	IV-6-B-121
IV-6-B-24	Routine RECONN	IV-6-B-127
IV-6-B-25	Routine RCONDT	IV-6-B-130
IV-6-B-26	Routine GETSTD	IV-6-B-140
IV-6-B-27	Routine MOVREC	IV-6-B-144
IV-6-B-28	Routine RECHOM	IV-6-B-149
IV-6-B-29	Routine RECEND	IV-6-B-154
IV-6-B-30	Routine RECDMP	IV-6-B-156
IV-6-B-31	Routine SHTDWN	IV-6-B-160
IV-6-B-32	Routine RECON1	IV-6-B-165
IV-6-B-33	Routine RECON2	IV-6-B-186
IV-6-B-34	Routine RECON3	IV-6-B-198
IV-6-B-35	Routine REC3B	IV-6-B-217
IV-6-C-1	Printed Output Samples of FILE12 and Sensor Redundancy	IV-6-C-2
IV-6-C-2	Sample Printed Output of Countermortar/Counter- battery Radar, Routine CCOLLF	IV-6-C-4
IV-6-C-3	Printed Output Sample of Routine MOVSEN	IV-6-C-7
IV-6-C-4	Printed Output Sample of Routine SASSMT	IV-6-C-9
IV-6-C-5	Routine RCONDT Printout Statements	IV-6-C-12
IV-6-C-6	MOVREC Print Statements	IV-6-C-13
IV-6-C-7	Printed Output Sample from Routines GETSTD and SHTDWN	IV-6-C-14
IV-6-C-8	printed Output Sample from RECON1	IV-6-C-16
IV-6-C-9	Printed Output Sample from Routine RECON2	IV-6-C-17
IV-6-C-10	Printed Output Sample from Routine RECON3	IV-6-C-22

# FIGURES (Continued)

## PART 3

<u>Number</u>		<u>Part</u>
IV-7-1	Ground Combat Model Flow Diagram	IV-7-3
IV-7-2	Unit Orientation and Movement	IV-7-6
IV-7-3	Weapon-Target Coverage Pattern for Laterally Offset Units	IV-7-8
IV-7-4	Ground Combat Model Information Flow Schematic	IV-7-13
IV-7-5	Multiple Unit Engagement	IV-7-16
IV-7-6	Unit Orientation Variables	IV-7-18
IV-7-7	Line of Sight Probability Versus Range (km) for $\bar{r} = 0.6$ km	IV-7-24
IV-7-8	Relationship Between the Engagement Frontage and the Variable $d'$	IV-7-27
IV-7-9	Coverage Pattern Variables, Detailed	IV-7-29
IV-7-10	Coverage Pattern Variables	IV-7-30
IV-7-11	Search Vector Geometry	IV-7-42
IV-7-12	Typical Variation of $\lambda$ versus $\phi$	IV-7-42
IV-7-13	Visual Acuity Versus Off Angle	IV-7-44
IV-7-14	Example of Zones Defined by Weapon/Ammunition Range Capabilities	IV-7-54
IV-7-15	Estimated Coordinates	IV-7-68
IV-7-A-1	Transport System Specifications	IV-7-A-3
IV-7-A-2	Transport Types	IV-7-A-4
IV-7-A-3	Target Dimensions	IV-7-A-5
IV-7-A-4	Weapon System Specifications	IV-7-A-8
IV-7-A-5	Sensor Specifications	IV-7-A-11
IV-7-A-6	Sensor Employment	IV-7-A-12
IV-7-A-7	Weapon Systems Sensor Utilization Work Sheet	IV-7-A-14
IV-7-A-8	Target Priority and Vulnerability	IV-7-A-17
IV-7-A-9	Line of Sight	IV-7-A-19
IV-7-A-10	Probability of Line of Sight versus Range	IV-7-A-21
IV-7-A-11	Horizon and Background	IV-7-A-23
IV-7-A-12	Table of Seasons and Codes	IV-7-A-24
IV-7-A-13	Ground Combat Constant Data Card Input	IV-7-A-25
IV-7-A-14	Ground Combat Data Deck Structure	IV-7-A-27
IV-7-B-1	Ground Combat Model Segment Structure	IV-7-B-1
IV-7-B-2	Routine GCMDT	IV-7-B-3
IV-7-B-3	Routine GCM	IV-7-B-12
IV-7-B-4	Routine GSETUP	IV-7-B-21
IV-7-B-5	Routine DNUMBR	IV-7-B-28
IV-7-B-6	Routine STRBAR	IV-7-B-31
IV-7-B-7	Routine GCFIRE	IV-7-B-34
IV-7-B-8	Routine GCTDET	IV-7-B-41
IV-7-B-9	Routines GCKILL and GCKLL1	IV-7-B-46
IV-7-B-10	Routine GPKILL	IV-7-B-49

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
IV-7-B-11	Routine GBKEEP	IV-7-B-52
IV-7-B-12	Routine MURFAC	IV-7-B-54
IV-7-B-13	Routine GLEG	IV-7-B-56
IV-7-B-14	Routine GCLAST	IV-7-B-58
IV-7-B-15	Routine GUPDAT	IV-7-B-63
IV-7-C-1a	Unit Geometry and Target Acquisition Sample Output From Ground Combat Model	IV-7-C-2
IV-7-C-1b	Unit Geometry and Target Acquisition Sample Output From Ground Combat Model	IV-7-C-4
IV-7-C-2	Engagement Results Sample Output From Ground Combat Model	IV-7-C-8
IV-7-C-3	Mortar Scheduling Sample Output From Ground Combat Model	IV-7-C-10
IV-8-1	DIVWAG Area Fire Model Macroflow	IV-8-2
IV-8-2	FIREDT Macroflow	IV-8-6
IV-8-3	Weapon Munitions Characteristics Table	IV-8-7
IV-8-4	AREAFR Macroflow	IV-8-8
IV-8-5	Target Unit Assessment Geometry	IV-8-10
IV-8-6	Typical Unit Band Density	IV-8-11
IV-8-7	Typical Unit Personnel Protection	IV-8-19
IV-8-A-1	Area Fire Model Data Base	IV-8-A-2
IV-8-A-2	Personnel Postures	IV-8-A-3
IV-8-A-3	Completed Lethal Area of Weapon/Munition Matrix	IV-8-A-6
IV-8-A-4	Weapon/Munition Index	IV-8-A-7
IV-8-A-5	Weapon Munitions Delivery Characteristics	IV-8-A-10
IV-8-A-6	Weapons/Munition Projectile Description	IV-8-A-13
IV-8-A-7	Weapons/Munition Lethal Area for Personnel	IV-8-A-15
IV-8-A-8	Weapons/Munition Lethal Area for Enemy Equipment	IV-8-A-17
IV-8-A-9	Personnel Posture	IV-8-A-19
IV-8-A-10	Personnel Protection	IV-8-A-22
IV-8-A-11	Area Fire Card Formats, Load Routine AFMLD	IV-8-A-25
IV-8-A-12	Area Fire Model Data Deck Structure	IV-8-A-26
IV-8-B-1	Routine FIREDT	IV-8-B-3
IV-8-B-2	Routine AREAFR	IV-8-B-11
IV-8-B-3	Routine AKILL	IV-8-B-25
IV-8-B-4	Routine OASSES	IV-8-B-27
IV-8-C-1	Area Fire Assessment Sample Printed Output	IV-8-C-2
IV-9-1	TACFIRE Model Macroflow	IV-9-2
IV-9-2	AFTFC1 Macroflow	IV-9-5
IV-9-3	Method of Attack Table	IV-9-10
IV-9-4	AFTFC2 Macroflow	IV-9-12
IV-9-A-1	TACFIRE Model Data Base	IV-9-A-2
IV-9-A-2	Matrix of TACFIRE Constant Data Requirements, Blue Force	IV-9-A-4
IV-9-A-3	TACFIRE Target Priority Table Completed, Blue Force	IV-9-A-6



# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
IV-9-A-4	Range Intervals for Target Priorities	IV-9-A-7
IV-9-A-5	TACFIRE Method of Attack Table Complete, Blue Force	IV-9-A-11
IV-9-A-6	TACFIRE Summary of Targets Priorities for Platoon/ Company	IV-9-A-14
IV-9-A-7	TACFIRE Summary of Target Priorities for Battalion/ Brigade	IV-9-A-15
IV-9-A-8	TACLD Target Priority Summary List	IV-9-A-15
IV-9-A-9	TACFIRE Target Range/Priority Card Format	IV-9-A-18
IV-9-A-10	TACFIRE Method of Attack Card Format	IV-9-A-19
IV-9-A-11	TACFIRE Target with Priority and Method of Attack Card Format	IV-9-A-20
IV-9-A-12	Weapon/Munition Combinations	IV-9-A-21
IV-9-A-13	TACFIRE Data Matrix and Card Formats	IV-9-A-23
IV-9-A-14	TACFIRE Constant Data Input Cards	IV-9-A-26
IV-9-A-15	TACFIRE Data Deck Structure	IV-9-A-27
IV-9-B-1	Routine OVER9	IV-9-B-2
IV-9-B-2	Routine AFTFC1	IV-9-B-7
IV-9-B-3	Routine AFTFC2	IV-9-B-18
IV-9-C-1	TACFIRE Model Mission Scheduling Sample Printed Output	IV-9-C-1
IV-10-1	Air Ground Engagement Model Mission Type	IV-10-2
IV-10-2	Air Ground Engagement Model Macroflow	IV-10-5
IV-10-3	Resource Allocation Submodel Logical Flow	IV-10-9
IV-10-4	Friendly Airspace Overflight Submodel Logical Flow	IV-10-14
IV-10-5	Schematic for Calculating Safe Point, Blue Aircraft Attacking Red Target	IV-10-15
IV-10-6	Determination of the Number of Air Defense Weapons Within Range	IV-10-20
IV-10-7	Engagement Results Table	IV-10-26
IV-10-A-1	Visibility Codes	IV-10-A-2
IV-10-A-2	Minimum and Maximum Aircraft Requirements for Mission Performance	IV-10-A-4
IV-10-A-3	Mission Types and Codes	IV-10-A-5
IV-10-A-4	Minimum Ammunition and Personnel Requirements for Mission Performance	IV-10-A-8
IV-10-A-5	Maximum Ammunition and Personnel Requirements for Mission Performance	IV-10-A-10
IV-10-A-6	Aircraft Mission Preparation Delay Time	IV-10-A-12
IV-10-A-7	Aircraft Flight Speeds (Knots)	IV-10-A-15
IV-10-A-8	Flight Area Codes	IV-10-A-14
IV-10-A-9	Aircraft Landing Time Requirements	IV-10-A-17
IV-10-A-10	New Mission Availability Time for Returning Aircraft	IV-10-A-20
IV-10-A-11	Degradation of Available Aircraft for Maintenance	IV-10-A-23
IV-10-A-12	Air Defense Weapon Characteristics	IV-10-A-25

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
IV-10-A-13	Reduced Effectiveness of Air Defense Weapons (Percentages)	IV-10-A-27
IV-10-A-14	Terrain and Vegetation	IV-10-A-30
IV-10-A-15	Terrain and Vegetation Effects on Air Defense Weapons	IV-10-A-31
IV-10-A-16	Altitude Codes	IV-10-A-32
IV-10-A-17	Aircraft Average Vulnerable Area Against Specific Weapon Types for Various Kill Categories	IV-10-A-34
IV-10-A-18	Engagement Results Table	IV-10-A-37
IV-10-A-19	Engagement Results of Air Strikes	IV-10-A-38
IV-10-A-20	Activity Types and Codes Versus Target Type	IV-10-A-39
IV-10-A-21	Example Engagement Results Input	IV-10-A-41
IV-10-A-22	Air Ground Engagement Card Format Listings	IV-10-A-43
IV-10-A-23	Air Ground Engagement Data Deck Structure	IV-10-A-44
IV-10-B-1	Routine AIRGND	IV-10-B-3
IV-10-B-2	Routine BTF	IV-10-B-8
IV-10-B-3	Routine ATB	IV-10-B-11
IV-10-B-4	Routine AVAIL	IV-10-B-17
IV-10-B-5	Routine PREPTM	IV-10-B-18
IV-10-B-6	Routine TIMES	IV-10-B-21
IV-10-B-7	Routine SELECT	IV-10-B-26
IV-10-B-8	Routine ENRATA	IV-10-B-30
IV-10-B-9	Routine TORA	IV-10-B-38
IV-10-B-10	Routine BTA	IV-10-B-46
IV-10-B-11	Routine SEG5	IV-10-B-51
IV-10-C-1	DSL Air Mission Request Output	IV-10-C-2
IV-10-C-2	Automatic Air Mission Request Output	IV-10-C-8
IV-10-C-3	En Route Attrition Printed Output from Air Ground Engagement Model	IV-10-C-10
IV-10-C-4	Engagement Results Printed Output from Air Ground Engagement Model	IV-10-C-18
IV-10-C-5	Safe Point Update Printed Output from Air Ground Engagement Model	IV-10-C-21
IV-10-C-6	Mission End Printed Output from Air Ground Engagement Model	IV-10-C-22
IV-11-A-1	Suppression Time Table (Seconds)	IV-11-A-2
IV-11-A-2	Suppression Group Table	IV-11-A-3
IV-11-A-3	Suppression Time Table	IV-11-A-4
IV-11-A-4	Suppression Model Data Deck Structure	IV-11-A-6
IV-11-B-1	Routine SUPRES	IV-11-B-2
IV-11-C-1	Routine SUPRES Sample Printed Output from Suppression Model	IV-11-C-2
IV-12-1	Nuclear Assessment Model Flow	IV-12-3
IV-12-2	Effect of Elliptical Dispersion Pattern on Damage Assessment	IV-12-16

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
IV-12-3	Effect of Error in Height of Burst on Error in Range	IV-12-18
IV-12-4	Biological Response to Nuclear Radiation	IV-12-28
IV-12-5	Barrier Assessment for a Moving Unit	IV-12-31
IV-12-A-1	Munition Type Data Form	IV-12-A-3
IV-12-A-2	Vulnerability Data Form	IV-12-A-6
IV-12-A-3	Personnel Postures, Nuclear Environment	IV-12-A-7
IV-12-A-4	Weapon Munition Characteristics	IV-12-A-9
IV-12-A-5	Nuclear Weapon Effects	IV-12-A-12
IV-12-A-6	Example of Effects Table	IV-12-A-14
IV-12-A-7	Nuclear Weapon Equipment Vulnerability	IV-12-A-16
IV-12-A-8	Nuclear Barrier/Facility Vulnerability	IV-12-A-18
IV-12-A-9	Nuclear Environment Personnel Posture	IV-12-A-20
IV-12-A-10	Nuclear Environment Personnel Posture for Stay Activity	IV-12-A-22
IV-12-A-11	Induced Radiation Barrier	IV-12-A-24
IV-12-A-12	Nuclear Assessment Model Constant Data Input Cards	IV-12-A-25
IV-12-A-13	Nuclear Assessment Model Data Deck Structure	IV-12-A-27
IV-12-B-1	Routine NAM	IV-12-B-3
IV-12-B-2	Routine NFIRE	IV-12-B-5
IV-12-B-3	Routine NFIRDT	IV-12-B-9
IV-12-B-4	Routine BURST	IV-12-B-11
IV-12-B-5	Routine NDRD	IV-12-B-13
IV-12-B-6	Routine SUPSCH	IV-12-B-15
IV-12-B-7	Routine ASUNIT	IV-12-B-17
IV-12-B-1	Routine HEMOVN	IV-12-B-19
IV-12-B-9	Routine UPRMPT	IV-12-B-21
IV-12-B-10	Routine NASPER	IV-12-B-23
IV-12-B-11	Routine NASEOH	IV-12-B-26
IV-12-B-12	Routine FNDLNK	IV-12-B-30
IV-12-B-13	Routine FNDLOS	IV-12-B-31
IV-12-B-14	Routine STUPRO	IV-12-B-34
IV-12-B-15	Routine CFUN	IV-12-B-36
IV-12-B-16	Routine BANCIR	IV-12-B-38
IV-12-B-17	Routine NBARAS	IV-12-B-41
IV-12-B-18	Routine BARUPE	IV-12-B-44
IV-12-B-19	Routine BARNUM	IV-12-B-45
IV-12-B-20	Routine NCUNIT	IV-12-B-47
IV-12-B-21	Routine BAMUNT	IV-12-B-50
IV-12-C-1	Routine NFIRE Output	IV-12-C-2
IV-12-C-2	Routine NBARAS Output	IV-12-C-7
IV-12-C-3	Routine NCUNIT Output	IV-12-C-8
IV-12-C-4	Routine BAMUNT Sample Printout	IV-12-C-10

# FIGURES (Continued)

## PART 4

<u>Number</u>		<u>Page</u>
IV-13-1	Movement Model Macro Flow	IV-13-2
IV-13-2	DSL Move Path Schematic	IV-13-4
IV-13-3	MOVESR Macroflow	IV-13-10
IV-13-4	MOVEDT Macroflow	IV-13-14
IV-13-5	Movement Rate Determination Schematic	IV-13-15
IV-13-6	MOVE Macroflow	IV-13-19
IV-13-A-1	Movement Model Data Base	IV-13-A-2
IV-13-A-2	Matrix of Travel Modes Required for Mobility Categories	IV-13-A-6
IV-13-A-3	Data Base Preparation Schematic	IV-13-A-15
IV-13-A-4	Example of Breach Force Decision Table	IV-13-A-17
IV-13-A-5	TRAVEL Mode and Mobility Category	IV-13-A-20
IV-13-A-6	Unit Type Designators by Mobility Category	IV-13-A-21
IV-13-A-7	Mobility Classification	IV-13-A-23
IV-13-A-8	Mobility Exclusion List	IV-13-A-25
IV-13-A-9	Unit Road Movement Rates	IV-13-A-27
IV-13-A-10	Vehicular Road Movement Rate	IV-13-A-29
IV-13-A-11	Unit Cross Country Movement Rate	IV-13-A-31
IV-13-A-12	Vehicular Cross Country Movement Rate	IV-13-A-34
IV-13-A-13	Vehicular Fuel Consumption Rates	IV-13-A-36
IV-13-A-14	Aircraft Fuel Consumption Rates	IV-13-A-38
IV-13-A-15	Breach/Force Decision Tables	IV-13-A-41
IV-13-A-16	Movement Model Constant Data Input Cards	IV-13-A-42
IV-13-A-17	Movement Model Data Deck Structure	IV-13-A-43
IV-13-B-1	Routine MOVESR	IV-13-B-4
IV-13-B-2	Routine MOVEDT	IV-13-B-9
IV-13-B-3	Routine OBSRCH	IV-13-B-12
IV-13-B-4	Routine MOVE	IV-13-B-16
IV-13-C-1	Printed Output Sample from Routine MOVE	IV-13-C-2
IV-13-C-2	Printed Output Sample from Routine MOVEDT	IV-13-C-3
IV-13-C-3	Printed Output Sample from Routine MOVESR	IV-13-C-6
IV-13-C-4	Routine OBSRCH Print Statements	IV-13-C-8
IV-14-1	Engineer TASK Activities	IV-14-2
IV-14-2	Engineer Driver Routine, ENGR	IV-14-9
IV-14-3	Engineer Priority Routine, EPRIOR	IV-14-11
IV-14-4	Engineer Task Priority Indicators for Start Tasks	IV-14-12
IV-14-5	Engineer Feasibility Routine, EFEASI	IV-14-16
IV-14-6	Engineer UPDATE Routine, EUPDAT	IV-14-20
IV-14-7	Engineer Release Routine, ERELEA	IV-14-24
IV-14-8	Engineer Nuclear Routine, ENUCLE	IV-14-26
IV-14-A-1	Barrier System Locations Card Format	IV-14-A-3
IV-14-A-2	Barrier Line Identification Card Format	IV-14-A-4

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
IV-14-A-3	Barrier Segment Details Card Format	IV-14-A-6
IV-14-A-4	Barrier Segment Mnemonics Applicable to Engineer Model	IV-14-A-8
IV-14-A-5	Task Identification Card Format	IV-14-A-12
IV-14-A-6	Task Performance Rates for Equipment and Supplies Card Format	IV-14-A-15
IV-14-A-7	Task Performance Rates for Troops Card Format	IV-14-A-19
IV-14-A-8	Engineer Task Rate Units	IV-14-A-21
IV-14-A-9	Task Performance Rates for Environment Card Format	IV-14-A-23
IV-14-A-10	Task Performance Rates for Vehicles Crossing Minefields Card Formats	IV-14-A-25
IV-14-A-11	Contingency Levels for Equipment Card Format	IV-14-A-27
IV-14-A-12	Contingency Levels for Troops Card Format	IV-14-A-28
IV-14-A-13	Unit Type Codes	IV-14-A-29
IV-14-A-14	Engineer Constant Data Card Types	IV-14-A-30
IV-14-A-15	Barrier and Facility Data Deck Structure	IV-14-A-31
IV-14-A-16	Engineer Task Data Deck Structure	IV-14-A-32
IV-14-B-1	Routine ENGR	IV-14-B-3
IV-14-B-2	Routine EPRIOR	IV-14-B-13
IV-14-B-3	Routine PROPTY	IV-14-B-16
IV-14-B-4	Routine EFEASI	IV-14-B-21
IV-14-B-5	Routine FESBIL	IV-14-B-25
IV-14-B-6	Routine ERELEA	IV-14-B-29
IV-14-B-7	Routine EUPDAT	IV-14-B-34
IV-14-B-8	Routine CREAL6	IV-14-B-39
IV-14-B-9	Routine MPRALC	IV-14-B-42
IV-14-B-10	Barrier Quadrature Example	IV-14-B-45
IV-14-B-11	Routine NUCSCH	IV-14-B-47
IV-14-C-1	Engineer Model Sample Output	IV-14-C-2
IV-15-1	Macroflow of Airmobile Operation	IV-15-10
IV-15-2	Macroflow of Accept Transport - Segment 1	IV-15-12
IV-15-3	Macroflow of Airmobile Assault - Segment 2	IV-15-16
IV-15-4	Macroflow of Release Transport - Segment 3	IV-15-22
IV-15-5	Schematic Concept of Forward Area Refueling and Rearming System	IV-15-27
IV-15-6	Illustrative Example of Refuel/Rearm Situation	IV-15-31
IV-15-7	Example of Refuel and Rearm Scheduling	IV-15-31
IV-15-8	Macroflow of In-flight Attrition Submodel	IV-15-35
IV-15-9	Macroflow of In-flight Attrition Submodel (Section 2)	IV-15-37
IV-15-10	Macroflow of In-flight Attrition Submodel (Section 3)	IV-15-39
IV-15-11	Horizontal Aspect Angle	IV-15-50
IV-15-A-1	Second Priority Equipment Items Card Format	IV-15-A-3
IV-15-A-2	Airmobile Mix Table Card Format	IV-15-A-4
IV-15-A-3	FRRA Definition Card Format	IV-15-A-7

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
IV-15-A-4	Refuel Device Definition Card Format	IV-15-A-8
IV-15-A-5	Suppression Mission Data (GTAA) Card Format	IV-15-A-10
IV-15-A-6	Acquisition Data (Card Type 1)	IV-15-A-13
IV-15-A-7	Acquisition Data (Card Type 2)	IV-15-A-14
IV-15-A-8	Aircraft Data Card Format	IV-15-A-16
IV-15-A-9	Air Defense Weapon Data (Card Type 1) Card Format	IV-15-A-18
IV-15-A-10	Air Defense Weapon Data (Card Type 2) Card Format	IV-15-A-21
IV-15-A-11	Air Defense Weapon Data (Card Type 3) Card Format	IV-15-A-23
IV-15-A-12	Air Defense Weapon Data (Card Type 4) Card Format	IV-15-A-25
IV-15-A-13	Missile P <sub>K</sub> Versus Miss Distance for Rotary Wing Aircraft Card Format	IV-15-A-27
IV-15-A-14	Missile P <sub>K</sub> Versus Miss Distance for Fixed Wing Aircraft Card Format	IV-15-A-29
IV-15-A-15	Airmobile Model Constant Data Input Cards	IV-15-A-31
IV-15-A-16	Airmobile Model Data Deck Structure	IV-15-A-32
IV-15-B-1	Routine AMBLEX	IV-15-B-2
IV-15-B-2	Routine ACCPT	IV-15-B-3
IV-15-B-3	Routine ACCPT1	IV-15-B-6
IV-15-B-4	Routine ACCPT2	IV-15-B-15
IV-15-B-5	Routine ACCPT3	IV-15-B-18
IV-15-B-6	Routine ASULT	IV-15-B-21
IV-15-B-7	Routine ASULT1	IV-15-B-23
IV-15-B-8	Routine ASULT2	IV-15-B-26
IV-15-B-9	Routine ASULT3	IV-15-B-31
IV-15-B-10	Routine ASULT4	IV-15-B-36
IV-15-B-11	Routine RREXEC	IV-15-B-43
IV-15-B-12	Routine RRFRRRA	IV-15-B-45
IV-15-B-13	Routine RRQUE	IV-15-B-52
IV-15-B-14	Routine RRFLD	IV-15-B-56
IV-15-B-15	Routine RRFUEL	IV-15-B-65
IV-15-B-16	Routine RATIM	IV-15-B-69
IV-15-B-17	Routine RRTURN	IV-15-B-71
IV-15-B-18	Routine RLEAS	IV-15-B-72
IV-15-B-19	Routine RLEAS1	IV-15-B-75
IV-15-B-20	Routine RLEAS2	IV-15-B-79
IV-15-B-21	Routine RLEAS3	IV-15-B-83
IV-15-B-22	Routine RLEAS4	IV-15-B-87
IV-15-B-23	Routine PENRAT	IV-15-B-89
IV-15-B-24	Routine SETTYP	IV-15-B-91
IV-15-B-25	Routine MISUNT	IV-15-B-93
IV-15-B-26	Routine CIRCLE	IV-15-B-96
IV-15-B-27	Routine GASRCH	IV-15-B-98
IV-15-B-28	Routine ATTDVR	IV-15-B-100
IV-15-B-29	Routine MES	IV-15-B-104
IV-15-B-30	Routine SIS	IV-15-B-115
IV-15-B-31	Routine GTAA	IV-15-B-123
IV-15-B-32	Routine LOSSES	IV-15-B-134

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
IV-15-C-1	Printed Output Sample from Routine ACCPT1	IV-15-C-2
IV-15-C-2	Printed Output Sample from Routine ACCPT2	IV-15-C-5
IV-15-C-3	Printed Output Sample from Routine ASULT1	IV-15-C-6
IV-15-C-4	Printed Output Sample from Routine ASULT2	IV-15-C-7
IV-15-C-5	Printed Output Sample from Routine ASULT3	IV-15-C-9
IV-15-C-6	Printed Output Sample from Routine RLEAS1	IV-15-C-10
IV-15-C-7	Printed Output Sample from Routine RLEAS2	IV-15-C-12
IV-15-C-8	Printed Output Sample from Routine RLEAS3	IV-15-C-13
IV-15-C-9	Sample Printed Output from Rearm/Refuel	IV-15-C-15
IV-15-C-10	Sample Printed Output from Routine SIS	IV-15-C-20
IV-15-C-11	Sample Printed Output from Routine GTAA	IV-15-C-24
IV-15-C-12	Sample Printed Output from Routine LOSSES	IV-15-C-28
IV-15-C-13	Sample Printed Output from Routine MES	IV-15-C-30
IV-16-1	Flow of Resupply Actions in Combat Service Support Model	IV-16-4
IV-16-2	Combat Service Support Model	IV-16-8
IV-16-3	Processing of Critical Consumables and Expendables	IV-16-10
IV-16-4	Basic Inventory Model Schematic	IV-16-11
IV-16-5	Transport Assignment Algorithm	IV-16-17
IV-16-6	Processing of Major End Items and Personnel	IV-16-22
IV-16-A-1	Combat Service Support Data Base	IV-16-A-2
IV-16-A-2	Elapsed Time to Handle Transports and Class I Consumption Card Format	IV-16-A-4
IV-16-A-3	Items of Supply by Class Card Format	IV-16-A-6
IV-16-A-4	Weights and Volumes of Transports Capacities Card Format	IV-16-A-9
IV-16-A-5	Class VII and Personnel Available for Resupply by Card Format	IV-16-A-11
IV-16-A-6	Combat Service Support Card Format Listings	IV-16-A-13
IV-16-A-7	Combat Service Support Data Deck Structure	IV-16-A-14
IV-16-B-1	Combat Service Support Model Initialization	IV-16-B-5
IV-16-B-2	Pending Supply Actions	IV-16-B-6
IV-16-B-3	Force Loop	IV-16-B-12
IV-16-B-4	Personnel and Major End Items Resupply	IV-16-B-21
IV-16-B-5	Routine TRNTIM	IV-16-B-26
IV-16-B-6	Routine EVTFIL	IV-16-B-29
IV-16-B-7	Routine SUPFIL	IV-16-B-31
IV-16-B-8	Routine BOFILE	IV-16-B-34
IV-16-B-9	Routine MISOUT	IV-16-B-36
IV-16-B-10	Routine RESLVE	IV-16-B-38
IV-16-B-11	Routine CSSANL	IV-16-B-40
IV-16-C-1	Routine SERSUP Sample Output	IV-16-C-1
IV-17-1	Special Purpose Routine Macroflow	IV-17-2

# FIGURES (Continued)

		<u>Page</u>
IV-17-B-1	Routine INITIAL	IV-17-B-2
IV-17-B-2	Routine COMMDP	IV-17-B-6
IV-17-B-3	Routine EXIT1	IV-17-B-8
IV-17-B-4	Routine EAT	IV-17-B-11
IV-17-C-1	Printed Output Sample from Routine INITIAL	IV-17-C-2
IV-17-C-2	Routine COMMDP Printed Output Sample	IV-17-C-5
IV-18-B-1	Routine DOSR	IV-18-B-3
IV-18-B-2	Routine DBSR	IV-18-B-6
IV-18-B-3	Routine TRUTH	IV-18-B-8
IV-18-C-1	DIVWAG Scenario Language Interface Printout	IV-18-C-2
IV-19-B-1	Routine INTSPT	IV-19-B-6
IV-19-B-2	Routine PONTLN	IV-19-B-8
IV-19-B-3	Routine CELLST	IV-19-B-10
IV-19-B-4	Routine LOS	IV-19-B-14
IV-19-B-5	Routine SEARCH	IV-19-B-18
IV-19-B-6	Routine INTPOL	IV-19-B-20
IV-19-B-7	Routine RANKA and RANK	IV-19-B-23
IV-19-B-8	Routine SCALE	IV-19-B-25
IV-19-B-9	Routine RANDU and RAND1	IV-19-B-28
IV-19-B-10	Routine IUIDF	IV-19-B-30
IV-19-B-11	Routine ADDUNT	IV-19-B-32
IV-19-B-12	Routine EOH2OT	IV-19-B-34
IV-19-B-13	Routine PUTOUT	IV-19-B-36
IV-19-B-14	Routine SCORE	IV-19-B-38
IV-19-B-15	Routine KPICK	IV-19-B-40



# FIGURES (Continued)

## PART 5

<u>Number</u>		<u>Page</u>
V-2-A-1	Routine REPCTL	V-2-A-2
V-3-B-1	Routine UPEOH	V-3-B-3
V-3-B-2	Routine DNCHAL	V-3-B-10
V-3-B-3	Routine USTAT1	V-3-B-14
V-3-B-4	Routine PUTDET	V-3-B-24
V-3-B-5	Routine EOHLOC	V-3-B-34
V-3-B-6	Routine EOHPER	V-3-B-39
V-3-B-7	Routine UACTR	V-3-B-44
V-3-B-8	Routine GET31	V-3-B-50
V-3-B-9	Routine SUM	V-3-B-52
V-3-C-1	Status and Activity Reports	V-3-C-2
V-4-B-1	Routine UID501	V-4-B-2
V-4-B-2	Routine PLREPT	V-4-B-5
V-4-B-3	Routine PLOVFL	V-4-B-6
V-4-C-1	Period Planning Report	V-4-C-1
V-5-B-1	Routine INTLRP	V-5-B-4
V-5-C-1	Period Intelligence Report	V-5-C-2
V-6-B-1	Routine ENGREL	V-6-B-2
V-6-C-1	Period Barrier Report	V-6-C-2
VI-1-1	Analysis Output Routine	VI-1-2
VI-1-2	Analysis Output Processor Macroflow	VI-1-3
VI-2-A-1	Input Card Format for PREP	VI-2-A-2
VI-2-A-2	Input Card Format for PTAPE	VI-2-A-5
VI-2-B-1	Routine PREP	VI-2-B-3
VI-2-B-2	Routine PTAPE	VI-2-B-8
VI-2-B-3	Routine UIDCD	VI-2-B-12
VI-2-B-4	Routine UTDCD	VI-2-B-14
VI-2-B-5	Routine CLSCD	VI-2-B-16
VI-2-B-6	Routine TIMCD	VI-2-B-18
VI-2-B-7	Routine DEVCD	VI-2-B-20
VI-2-B-8	Routine ENDCD	VI-2-B-22
VI-2-B-9	Routine DRSPLT	VI-2-B-28
VI-2-B-10	Routine WRTDOR	VI-2-B-32
VI-2-C-1	Printed Output Sample of Routine PREP	VI-2-C-2
VI-2-C-2	Printed Output Sample of Routine UXR	VI-2-C-3
VI-2-C-3	Printed Output Sample of Routine PTAPE	VI-2-C-5
VI-3-A-1	Input Card for PHIST	VI-3-A-2
VI-3-B-1	Routine PHIST	VI-3-B-2
VI-3-B-2	History Record Print Routines	VI-3-B-4
VI-3-C-1	Area Fire Assessment Record	VI-3-C-2

# FIGURES (Continued)

<u>Number</u>		<u>Page</u>
VI-3-C-2	Area Fire Event Record	VI-3-C-3
VI-3-C-3	Air Ground Engagement Mission Request Record	VI-3-C-5
VI-3-C-4	Air Ground En Route Attrition Record	VI-3-C-6
VI-3-C-5	Air Ground Target Engagement Record	VI-3-C-8
VI-3-C-6	Air Ground Target Assessment Record	VI-3-C-9
VI-3-C-7	Ground Combat Attacker-Defender Pair Record	VI-3-C-11
VI-3-C-8	Ground Combat Battle Unit Record	VI-3-C-13
VI-3-C-9	Back Order Event Record	VI-3-C-15
VI-3-C-10	Supply Actions	VI-3-C-16
VI-3-C-11	Supply Action Event Record	VI-3-C-17
VI-3-C-12	Movement Event Record	VI-3-C-21
VI-4-A-1	Input Cards for ANCARD	VI-4-A-5
VI-4-B-1	Routine ANCARD	VI-4-B-4
VI-4-B-2	Routine GCMOD	VI-4-B-14
VI-4-B-3	Routine MATOUT	VI-4-B-22
VI-4-B-4	Routine MPRCES	VI-4-B-24
VI-4-B-5	Routine AGM	VI-4-B-36
VI-4-B-6	Routine AFM	VI-4-B-41
VI-4-B-7	Routine IOMAT	VI-4-B-47
VI-4-B-8	Routine SETUP	VI-4-B-52
VI-4-C-1	Printed Output of Routine ANCARD	VI-4-C-2
VI-4-C-2	Printed Output from Routine AFM	VI-4-C-3
VI-4-C-3	Printed Output from Routine AGM	VI-4-C-3
VI-4-C-4	Printed Output from Routine GCMOD	VI-4-C-5
VI-5-A-1	Update Data Card for Routine MTXUP	VI-5-A-2
VI-5-A-2	Input Card for Routine ANALYS	VI-5-A-4
VI-5-B-1	Routine MTXMRG	VI-5-B-2
VI-5-B-2	Routine MTXUP	VI-5-B-6
VI-5-B-3	Routine ANALYS	VI-5-B-10
VI-5-B-4	Routine CALCH	VI-5-B-15
VI-5-B-5	Routine PAIR	VI-5-B-17
VI-5-B-6	Routine FRDMAN	VI-5-B-22
VI-5-B-7	Routine DSESRT	VI-5-B-24
VI-5-B-8	Routine FRGFIT	VI-5-B-26
VI-5-B-9	Routine ANAOVA	VI-5-B-29
VI-5-C-1	MTXMRG Output	VI-5-C-2
VI-5-C-2	MTXUP Output	VI-5-C-3
VI-5-C-3	ANALYS Output	VI-5-C-5
VII-3-B-1	Routine CREATE	VII-3-B-3
VII-3-B-2	Routine REMOVE	VII-3-B-6
VII-3-B-3	Routine ADDRCD	VII-3-B-10
VII-3-B-4	Routine ADDWRD	VII-3-B-14
VII-3-B-5	Routine SUBRCD	VII-3-B-18
VII-3-B-6	Routine SUBWRD	VII-3-B-22
VII-3-B-7	Routine FILE	VII-3-B-26

FIGURES (Continued)

<u>Number</u>		<u>Page</u>
VII-3-B-8	Routine RECORD	VII-3-B-29
VII-3-B-9	Routine WORD	VII-3-B-34
VII-3-B-10	Routine GETPUT	VII-3-B-37
VII-3-B-11	Routine TRNSMT	VII-3-B-40
VII-3-B-12	Routine FSL	VII-3-B-47
VII-3-B-13	Routine SHFTDN	VII-3-B-49
VII-3-B-14	Routine MASSIO	VII-3-B-53
VII-4-B-1	Routine IOTERN	VII-4-B-2
VII-4-B-2	Routine IOWETH	VII-4-B-6
VII-4-B-3	Routine UTILLD	VII-4-B-8
VII-4-B-4	Routine UTILDP	VII-4-B-11
VII-4-C-1	Utility LOAD Routine Printed Output	VII-4-C-2
VII-4-C-2	Utility DUMP Routine Printed Output	VII-4-C-2
VII-5-B-1	Routine MVCHAR	VII-5-B-2
VII-5-B-2	Routine KHAR	VII-5-B-4
VII-5-B-3	Routine PACK28	VII-5-B-6
VII-5-B-4	Routine INTCHR	VII-5-B-8
VII-5-B-5	Routine SNATCH	VII-5-B-10

NOTES

SECTION I

DIVWAG SYSTEM GENERAL DESCRIPTION

*12-11-55*

## CHAPTER 1

### INTRODUCTION

1. PURPOSE. The purpose of this volume is to provide computer programmers and military operation research analysts with information necessary to understand the detailed operation of the DIVWAG system, the means by which the models represent the military activities simulated, and the means by which the constituent computer programs perform the necessary model functions. This volume also provides the instructions for preparing the DIVWAG data base.

2. SCOPE. This volume includes the following elements of the DIVWAG system documentation.

- . Descriptions of the mathematical and logical processes
- . Input data requirements
- . Detailed program descriptions
- . Output data descriptions
- . Data storage techniques
- . Diagnostic message handbook
- . Computer system requirements.

3. ORGANIZATION. This volume is divided into eight sections. The introductory section provides a brief description of the DIVWAG system and its developmental history followed by the minimum computer system requirements and definitions of terms. The second through sixth sections describe the Constant Data Input, Orders Input, Period, Period Output, and Analysis Output Processors respectively. Each major element of a processor constitutes a chapter with its associated appendixes. The mathematical and logical processes are described in the chapter and the appendixes include the input requirements, detailed program descriptions, output descriptions, and references. Program source listings are available under separate cover. Section seven describes utility routines used by more than one processor. A handbook, listing and describing all diagnostic messages, is included in section eight.

NOTES

*Re-*



## CHAPTER 2

### MODEL DESIGN AND DEVELOPMENT

#### 1. NEED FOR A DIVISION WAR GAME MODEL:

a. Within the Army Planning System,<sup>1</sup> there are two significant planning tasks that demand the use of a credible and acceptable division war game model: objective force and resource planning, and approved (constrained) force planning. Both planning tasks contribute to the primary objectives of the Army Planning System; namely, to:

- . Provide timely, pertinent Army views for consideration by the Secretary of Defense.
- . Contribute persuasively to the formulation and presentation of joint military strategy, force objectives, and other matters of the Joint Strategic Planning System.
- . Provide timely guidance to Army staffs and commanders.

b. Annually, as part of the Department of Defense Planning, Programming, and Budgeting System (PPBS), the Military Department staffs submit to the Secretary of Defense plans for those Service forces and resources they feel are necessary to meet the national security objective. These plans are referred to as objectives or requirements plans. Imbedded in these plans are the results of detailed analyses, studies, and games that supply the rationale to justify the major Service forces planned or recommended. As an integral part of the preparation of Army objectives plans, detailed study and analysis is devoted to cost-versus-effectiveness trade-offs among alternative forces to achieve the desired objective of countering the enemy threat to the United States and its Allies. The most significant building block in advancing the Army's recommendations is the division or, more precisely, the division force equivalent (DFE). A division war game model, properly applied, will provide evaluation data to support Army objective force planning.

c. In the course of exercising the PPBS, the Secretary of Defense makes an annual decision on the apportionment of resources to each Military Department, and the results of the decision are embodied in the Five-Year Defense Program (FYDP). With this decision, the resources available to each Military Service are approved, as are the forces represented by the resources. The Army, as with the other Services, enters into a period of constrained force planning. An obvious objective of such an exercise is attaining the most combat effective force for the resources available. Again, cost effectiveness studies and analyses contribute to those approved force structures appearing in the Army Force Development Plan; and, as with objective force planning,

---

1. AR 1-1

the division or the division force equivalent becomes the basic building block. A division war game model is of immeasurable value in arriving at the design and structure of the Army in the field.

d. Aside from direct application of a division war game model to the formalized Army Planning System, there is a major requirement to assess the value of competing individual systems (firepower; mobility; command, control, and communications; combat service support; and intelligence) to the overall combat effectiveness of Army forces in the field. All aspects and capabilities of these individual competing systems are studied, analyzed, and tested; however, until the systems are aggregated into a combat organization and evaluated as part of a composite system, the contribution (value) of each system to overall combat effectiveness cannot be properly assessed. A division war game model simulates all functions of land combat at an organizational level that permits evaluation of component organizations and systems.

## 2. DIVISION WAR GAME MODEL DEVELOPMENT:

a. During the late 1960s a war game model, DIVTAG II (DIVision Through Army Group), was developed for U.S. Army Combat Developments Command (USACDC) and exercised under the sponsorship of the USACDC Institute of Combined Arms and Support (USACDCICAS). DIVTAG II was recognized as a major developmental effort incorporating several potentially significant features, including the DIVTAG Scenario Language (DSL), which allowed gamers to formulate their game period plans and orders in English-like commands for communication to the DIVTAG II computer submodels.

b. Experience in use of the DIVTAG II model to generate game data for analysis determined that the DIVTAG II system--including both human and computer elements--functioned far too slowly to be an effective tool for responding to USACDC force analysis requirements. In addition, several flaws were suspected in the application of DIVTAG II algorithms to the simulation of combat reality.

c. On 10 August 1970, the Combat Developments Research Office of Computer Sciences Corporation (CSC-CDRO) initiated Project 3-71, Improvement of the ICAS War Game Model (IMPWAG) under Contract DAAG11-70-C-0875. The overall tasks of this project were to identify deficiencies and problem areas in DIVTAG II through a review of the documentation; to establish a priority list for correction/resolution of the deficiencies and problem areas; to correct or improve DIVTAG II within project constraints; and to document corrective action, improvements, and resolutions, as well as recommended follow-on corrective action. This project was completed on 15 May 1971.

d. In February 1971, project tasking was issued to CSC-CDRO to design, develop, and validate:

(1) A computer-assisted war game model(s) which would permit determination of the impact on force effectiveness of changes in the mixes of major weapons and other systems at an appropriate level of resolution.

1 APR 1976

(2) An analytical methodology for analyzing the output of the model(s) developed and determining the effectiveness of a single force.

(3) An analytical methodology for comparing alternative forces.

The project described above is entitled "Development of a Division War Game Model (DIVWAG)" and was completed on 31 December 1971.

e. On 15 November 1971, CSC-CDRO initiated Project 2-72, Improvement of the War Gaming Capability (WAGCAP), the objectives of which were to:

- . Incorporate selected improvements and additions into the DIVWAG Model
- . Train a government team in the major aspects of the application of the model to include associated analytical methodologies.

The project was completed on 15 July 1972.

f. The DIVTAG II model and its successor, the DIVWAG model, were programmed to execute on the Control Data Corporation 3300 computer system at Fort Leavenworth, Kansas. They contained numerous machine-dependent programming features. In July 1972, tasking was initiated under Project 1-73 to reprogram the DIVWAG model to conform with newly established U.S. Army software development standards. The model would thus be readily adaptable to many computer systems available to the government, and fully operational on the Control Data Corporation 6500 computer system installed at Fort Leavenworth, Kansas, in November 1972. The project included fully documenting the DIVWAG model and performing benchmark validation.

g. The programming project was completed in April 1973 and the government accepted the model from CSC on 15 April 1973. Since then, it has been utilized to support six study efforts:

- (1) Conceptual Armored Division (CONAR) Study (May 73 - Sep 73).
- (2) Family of Scatterable Mines (FASCAM) Study (December 73 - August 74).
- (3) TOW-DRAGON Antitank Mix Study (October 74 - December 74)
- (4) Integration of Intelligence From All Sources (IIFAS) Study (May 75 - Sep 75).
- (5) Antiarmor Systems Program Review (ASPR) (Jan 76 - Apr 76).
- (6) Legal Mix V Study (Apr 76 - Aug 76).

h. In August 1974 a contract was awarded to Braddock-Dunn-McDonald (BDM) to improve the Engineer and Movement models. The \$65,000 effort was completed in October 1975. Successful integration of the changes has not been made and the documentation does not reflect BDM's modifications.

NOTES

12

## CHAPTER 3

### SUMMARY DESCRIPTION

1. **MODEL OBJECTIVE.** The DIVWAG Model was developed as a computer-assisted war gaming system for use in simulating military interactions between opposing division-size forces and their major elements, with outputs permitting evaluation and comparison of the combat effectiveness of such division forces. The DIVWAG Model objective is to provide means for determining the impact on force effectiveness of changes in the mixes of major weapons and other systems. The DIVWAG Model permits two-sided simulations. The games using the DIVWAG Model can be open, semi-open, or closed. The games will be basically rigid, but the model can be operated with semi-rigid intelligence and special weapons assessment. Resolution is partially adjustable. Time resolution may be as small as a hundredth of a minute (0.01 minute), space resolution as small as one meter, and unit resolution as small as one individual; however, for practical gaming purposes, unit resolution is on the order of a battalion, depending on terrain scale and size forces being gamed.

2. **MODEL CAPABILITIES.** The model capabilities are discussed in terms of operational capabilities and operational scope.

a. **Operational Capabilities.** The DIVWAG Model has the following operational capabilities:

(1) Producing data for use in evaluating effectiveness of forces composed of maneuver units and their associated combat support and combat service support units.

(2) Producing detailed quantitative data for use in comparing the effectiveness of the alternative forces.

(3) Simulating high and mid-intensity conflict (nuclear and conventional war).

b. **Operational Scope.** The DIVWAG Model scope is defined in terms of command levels, military services, types of operations, geographical areas of operations, rate of operation, and combat functions.

(1) **Command Levels.** The DIVWAG Model has been designed to produce data to evaluate division forces; i.e., a division plus an appropriate slice of corps or field army troops. There is explicit representation of the task organization of the forces. Echelons of command are defined, and reports are prepared that reflect the status of each command echelon and include the aggregated status of all subordinate units. Communications are simulated between division, brigade/regiment, and battalion command units. A total of 1000 units can be played, divided among Red and Blue forces. These units must be structured into task organizations for each force.

(2) Military Services. The DIVWAG Model has been designed to accommodate and integrate all types of land forces (e.g., armored, mechanized, and airmobile) and supporting tactical air forces.

(3) Types of Operations. The DIVWAG Model provides for representation of the major types of military operations; i.e., offensive, defensive, retrograde (delay/withdrawal), covering force, and movement. In addition the model is capable of simulating high and mid-intensity conflict (nuclear and conventional war).

(4) Geographical Area of Operations. The DIVWAG Model can accommodate rectangular geographical areas as large as 8,000 kilometers on a side. The system is limited in its application to worldwide geography only by availability of appropriate input data.

(5) Rate of Operation. The rate of operation of the DIVWAG war game is not firmly established. It is estimated that the simulation model can produce game period turnaround data in a 2:1 ratio of computer time to combat time simulated for a relatively straightforward combat situation at a battalion level of resolution. Actual timing is a function of the level of resolution being gamed, the number of units being gamed, the level of military activity portrayed, and the nature of other computer jobs on the system when the model is being used in a multiprogramming environment. Considering the time required for period turnaround, a real time to game time pace of approximately 5:1 is considered reasonably attainable for gaming at a sufficient level of complexity to provide useful results.

(6) Combat Functions. The DIVWAG Model can address the following functions and evaluate the contribution to force effectiveness of varying the mixes of related elements:

- (a) Intelligence functions; surveillance and target acquisition.
- (b) Command, control, and communications functions; decision and communications delay times.
- (c) Firepower functions.
- (d) Mobility functions; aerial, ground, and firepower mobility.
- (e) Combat service support functions; supply and transportation; loss, expenditure, and consumption rates; personnel replacement.

3. FUNCTIONAL COMPONENTS AND CAPABILITIES. Functionally the DIVWAG Model is a dual system; it functions physically/electronically as a data processing system and, at the same time, it functions in simulation as a military combat system. To be complete, a description of the model must consider both aspects. The DIVWAG Model is described herein in terms of its data processing functional components and its military simulation functional capabilities.

a. Data Processing Functional Components. The DIVWAG software is divided functionally into five processors that communicate with each other through common files and records. Designations and functions of these processors are:

(1) Constant Data Input Processor. The Constant Data Input Processor receives data on cards, edits the data, and assembles the data onto tape and disk files.

(2) Orders Input Processor. The Orders Input Processor receives player operational orders in semi-military language and processes these orders into detailed instructions to the units simulated.

(3) Period Processor. The Period Processor receives translated player orders and simulates the military action.

(4) Period Output Processor. The Period Output Processor receives results from the Period Processor and compiles specific reports and gross summary reports.

(5) Analysis Output Processor. The Analysis Output Processor receives detailed data from the Period Processor as period history tapes; retrieves, arrays, and performs the statistical analysis of the data; and outputs the arrayed data in specified formats.

(6) Data Flow. The flow of data between processors and to the gaming staff and analysis team is displayed graphically in Figure I-3-1. The input and output of each processor are tabulated in Figure I-3-2.

b. Military Simulation Functional Capabilities. The Period Processor simulates an extremely broad and flexible spectrum of military activity through four categories of models (intelligence and control, firepower, mobility, and combat service support). The models are described individually in the following subparagraphs.

(1) Intelligence and Control (INC). This model provides the quantitative data necessary for evaluation of the contribution of sensor mixes to force effectiveness. It integrates the closely related functions of surveillance; target acquisition; combat intelligence; and command, control, and communications. Gamers are permitted to input intelligence from sources not simulated by model components. The information obtained from sensors and from gamer input is processed and used automatically by the Intelligence and Control Model to make requests for fire support on acquired targets. Fire missions are requested from available attack helicopters, Air Force close air support (CAS), or ground-based artillery by use of a set of decision rules, according to the situation. Sensor information is also converted into general intelligence by this model to produce a summary report at the end of each game period. The summary outlines the current status of what may be known at division level concerning the size, type, and location of the enemy forces in the battle area. This report is to aid the gamer in preparing orders for the next game period. The Intelligence and Control Model consists of three interrelated submodels: Collection, Processing, and Decision. The military functions simulated by the Intelligence and Control Model are summarized below and include:

(a) Sensing and Reporting. The capabilities of individual ground and aerial sensors are considered to simulate the detection and collection of information or intelligence on units of the opposing force and the summarizing

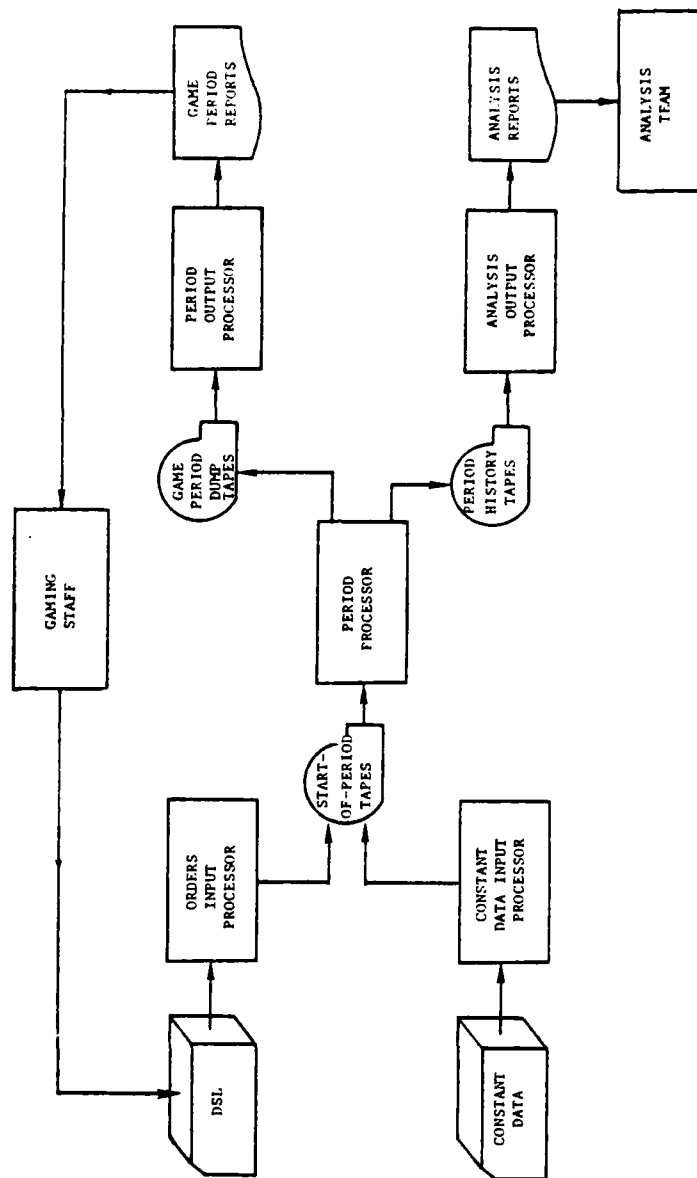


Figure I-3-1. DTWAG System Data Flow



Processor	Input	Output
Constant Data Input Processor	Constant Data Cards	Formatted dumps of data files.
Orders Input Processor	DSL Order Cards	Unit and battle order tables.
Period Processor	Start-of-Period Tapes	Game Period History Tapes, Game Period Dump Tapes, printed output from the processor.
Period Output Processor	Game Period Dump Tapes	Game Period Status, Activity, Intelligence and Barrier Reports.
Analysis Output Processor	Period History Tapes	Analysis Reports

Figure I-3-2. Processor Input and Output

of such information into sensing reports, which enter the intelligence chain. Both target and nontarget intelligence are simulated. Type sensors modeled include moving target indicator radar, unattended ground sensor fields, counter mortar and counterbattery radar, air defense radar, visual observers in light observation helicopters and fixed wing reconnaissance aircraft, surveillance aircraft (Mohawk type) with moving target indicator radar capability, and high performance reconnaissance aircraft with visual and various photographic capabilities.

(b) Time Delays. The model introduces separate delays for time consumed in each of the principal steps: intelligence collection, intelligence analysis, routing, and use of the information in decisions.

(c) Development of Targets for Fire Missions. A separate channel for development of target intelligence is simulated by the model. This channel provides acquired targets for fire missions without the relatively long delays involved in general intelligence processing.

(d) Intelligence Analysis and File Maintenance. Comparison of a new report with information already in the intelligence files is simulated, and if reports relate to the same unit they are consolidated. The existence of new units or parent units can be deduced. Intelligence analysis centers are simulated for each unit at maneuver battalion, brigade or regiment, and division levels. Files are designed to stay within the limits of 10, 20, and 100 reports, respectively, for these three echelons.

(e) Decisions on Information/Intelligence Flow and Requests for Fire Support. The routing of information or intelligence among intelligence analysis centers and command elements at the three echelons is simulated with the use of a flow structure and set of routing criteria, according to the information in each report. A similar set of criteria is used to determine whether a target qualifies for fire support and what type of fire support (attack helicopter, TACAIR, ground-based artillery) will be requested.

(f) Contents of Intelligence Report. The end-of-period contents of the division intelligence file are used for this report. Each report that met criteria to reach this file and was not discarded from the file in favor of a more recent record or consolidated into a record of a parent unit is reflected in the Intelligence Report. Items of estimated information given in the report for each opposing unit in this intelligence file include size, activity, type, direction of last move, time last sensed, and number of sensings attributed to this unit.

(2) Firepower. The firepower models provide the quantitative data necessary for evaluation of force effectiveness as a function of changes in mixes and types of major weapon systems. The models integrate all aspects of mid or high intensity combat where interaction of opposing forces may occur and result in personnel or materiel losses. The firepower models coordinate and integrate the effectiveness of combined arms teams by modifying assessment routines for high attrition events to account for concurrent and parallel killing capabilities. For example, when units are engaged in combat, kills are

related to the number of fire elements on opposing sides. Should an attack helicopter fire team make an attack during a ground combat cycle, the ground combat cycle will be interrupted immediately prior to the helicopter attack, the status of the units in ground combat will be updated to that moment, the effects of the helicopter attack on the current ground combat unit's status will be assessed, and the ground combat cycle will be resumed. The same assessment technique is followed for other interfacing firepower activities. Five models simulate the firepower function: Ground Combat, Area Fire, TACFIRE, Nuclear Assessment, and Air Ground Engagement. Each of these models assesses damage inflicted and produces loss, expenditure rate, and consumption data for use in evaluating the supply and transportation systems.

(a) Ground Combat Model:

1. The Ground Combat Model represents the interaction between the direct fire weapons of opposing maneuver units engaged in ground combat.

2. The model represents the interaction and the effects of weapons of cross-reinforced units. Combat power may be enhanced by employing combined-arm forces against the enemy. The effectiveness of the maneuver unit is largely dependent on the combination and coordination of weapon systems within the unit. The distance of separation of weapon systems is limited so that mutual support is possible when weapon density permits.

3. The impact of the environment is represented by the model. All movement in ground combat is subject to the constraints imposed by the environment wherein ability to move forces by ground is degraded by the effects of adverse weather, terrain, and visibility. The application of firepower is largely controlled by the environment since effectiveness of each weapon system is limited by its associated target acquisition capabilities. Target acquisition cannot occur unless line of sight exists between the observer and target. Line of sight may be severely limited due to terrain roughness, vegetation, and forestation. A firer may lose line of sight on a moving target before firing a round. A moving target may drop out of line of sight during the time of flight of the round. Target acquisition is limited by visibility, whether due to adverse weather or night combat operations. Under conditions of reduced visibility, target acquisition is enhanced by the employment of night vision equipment.

4. The interaction of each maneuver unit with the enemy is considered by the model in terms of a maneuver unit's effectiveness and vulnerability. The maneuver unit's effectiveness is influenced by the level of activity. As the level of activity increases, more weapon systems can acquire targets. As individual moving weapon systems stop to fire, the signature (i.e., evidence of that weapon firing) increases with the level of activity. The maneuver unit's vulnerability is influenced by the level of activity. A firing weapon system may disclose its position and become a target for enemy fire.

5. The Ground Combat Model relies heavily on the existence of data to describe weapon/ammunition effectiveness against varying target

15 APR 1975

types in a combat situation. The model also requires adequate data to describe the target acquisition capabilities of all employed sensor types other than unaided vision.

(b) Area Fire and TACFIRE Models. The Area Fire and TACFIRE Models simulate the scheduling of nonnuclear munitions for area fires, and the delivery and assessment of results of nonnuclear fires. Area fire events are generated by two methods. First, gamers issue fire orders prior to the engagement period for fire with specific ammunition at specific coordinates. Second, during the engagement period target information is developed by the Intelligence and Control Model with a request for nonnuclear artillery fire, and the TACFIRE Model automatically schedules the required fire missions for the fire units. The automatic mode is referred to as the TACFIRE mode. Targets in this mode consist of targets of opportunity and are limited to those enemy units detected and processed by the Intelligence and Control Model. Fire units are battalion or battery size, and integral fire units are used in attacking area fire targets. The fire units are constrained by range limitations, volley firing times, number of tubes or rails per fire unit, and weapons/munitions availability. A target threat priority value ranging from 1 to 9 is assigned to each area fire target, and is based upon its estimated size, type activity, range, and the tactical doctrine used in artillery employment for the particular game. Priority one is a higher priority than priority two, etc. If a backlog of targets exists, targets are engaged in highest priority order. The Area Fire and TACFIRE routines are separated into three functional classes: scheduling, delivery, and assessment. Most of the routines for the automatic TACFIRE mode are concerned with scheduling of fires. The delivery routines deliver the munitions on target and determine units whose presence in the impact area will require assessments. The assessment routines then calculate the effects of the fire events, based on number of rounds fired, lethal area of nonnuclear munitions, dimensions of the target, number and density of target elements, and target vulnerability. The assessment routine makes adjustments in target personnel and equipment to reflect losses and in the fire unit's munitions on hand to reflect expenditures.

(c) Air Ground Engagement Model. The Air Ground Engagement Model simulates for both opposing forces all air-to-ground and ground-to-air interactions falling within the definition of close air support (CAS) and otherwise directly related to ground combat operations. These operations include aircraft fires provided by other Services, and Army aircraft delivering direct aerial fires (DAF). The Air Ground Engagement Model determines all attrition and casualty results of such interactions. The Air Ground Engagement Model is sufficiently flexible that major changes in aircraft characteristics, quantity or mixes of the major weapon systems, or their modes of employment will be reflected in the measures of force effectiveness. Single or multiple aircraft flights are generated by the Intelligence and Control Model or are directed by gamer orders. Attrition of aircraft while in flight is based on the location of air defense capable units; i.e., units that contain air defense weapons. The Air Ground Engagement Model divides the flight path into the following segments as appropriate: air base to safe point, safe point to target, target to safe point, and safe point to air base. The Air Ground Engagement Model:

- . Selects from available aircraft and munitions types those best suited for the mission, determines the time required for aircraft preparation and pilot briefing, and schedules the time for aircraft to be airborne.
- . Maintains a current status record for aircraft assigned to a mission, to include munitions and fuel, aircraft losses caused by enemy activity, and effects of the aircraft on enemy targets.
- . Moves the aircraft progressively along the mission segments, assessing aircraft status,<sup>1</sup> attrition, and accomplishments at the completion of each segment to determine if the mission should continue.
- . Determines the results of attacks on targets in terms of aircraft losses and target losses.
- . Upon completion of the mission and return to the airbase, aggregates total mission results (including total mission time), assesses aircraft damage, and determines delay times for subsequent mission availability of the aircraft.

(d) Suppression Model. The treatment of suppressive effects of area fires or aerial strikes upon a unit was introduced to the DIVWAG Model through the addition of a Suppression Model. This model represents suppressive effects by the interruption of selected activities (unit movement, delivery of area fires, delivery of air defense fires) in response to incoming fire. Length of interruption depends on the activity interrupted and nature of fire received, and the interruption is extended as fires continue to be received.

(e) Nuclear Assessment Model. The Nuclear Assessment Model simulates the delivery of and the assessment of results of tactical nuclear fires. All nuclear fires are conducted in response to gamer fire order, issued prior to the simulated engagement period. The gamer fire order specifies the unit to fire, weapon and munition to fire, yield, height of burst (if controllable), and designated ground zero. Thus, all planning of nuclear fires must be carried out by the gamer prior to an engagement period. In response to a nuclear fire order, the Nuclear Assessment Model simulates the actual firing, the nuclear detonation, and the assessment of effects against all units as well as on obstacles and facilities within the effects area of the round. The detonation and effects of atomic demolition munitions (ADM) are also simulated by the model. Simulated effects include those due to blast, prompt nuclear and thermal radiation, and delayed effects due to induced radiation. Fallout effects are not simulated.

---

1. Includes effects of air defense activities, weather, and terrain.

(3) Movement Model. The Movement Model represents unit movement other than airmobile operations including the effects of those activities that serve to improve or impede movement. The Movement Model provides the quantitative data necessary for evaluation of force effectiveness as a function of ground movements, and the related effects of significant changes in the mixes and types of mobility means. The Movement Model considers the following aspects of force mobility:

(a) Air Movement. The Air Movement section of the model simulates moves by aircraft not in connection with airmobile operations. Air movements may be ordered externally by gamers or generated internally by the Air Ground Engagement Model. Aircraft availability, class IIIA supplies, and weather limitations are checked before the air movement is allowed. Air routes, altitudes, and speeds are specified by the gamer order or are determined by the model generating the movement internally. Once the air movement is initiated, it will be completed unless terminated due to losses. At the end of each flight segment the unit will be updated to reflect losses of aircraft and personnel and the status of associated supplies.

(b) Ground Movement. The Ground Movement section of the model simulates moves by surface transportation. Ground movements of units not engaged in combat are ordered by gamers. Ground movements are affected by category of move, unit mission, formation type, vehicle mobility characteristics, terrain conditions, daylight or darkness, road nets, weather, natural obstacles, and enemy created conditions. The maximum movement rate for a unit is the rate of the slowest type vehicle in the unit. Some of the other characteristics of ground movement are indicated below.

1. Administrative/Supply Movements. Administrative routes are generated by gamer orders; supply routes, by the Combat Service Support Model. The road movement rate depends on road type, grade, weather conditions, and nighttime or daytime conditions. Administrative movements are executed in segments determined by terrain cell boundaries or en route obstacles. Units are halted by events scheduled for the moving unit and by encountering obstacles. After the delay, the unit is not able to make up this lost time but continues to move at its appropriate rate.

2. Tactical Movements. Tactical routes are generated by gamer orders and are executed in segments in the same manner as administrative movements. Starting times and normal tactical movement rates are specified for each unit type for attack, withdrawal, and reinforcing missions, as well as for day or night movements. Units are halted for obstacles or minefields. After such delay, tactical units attempt to make up the lost time, and move at the limiting mobility class rate for that purpose. The limiting mobility class rate depends on terrain roughness and vegetation, slope and soil trafficability, forestation, weather conditions, and nighttime or daytime conditions. Each equipment type is assigned to a mobility class, and only those mobility classes used during tactical movements are considered for determining the limiting mobility class.

3. Maneuver Movements. Maneuvering weapon systems execute their movements at maximum limiting mobility class rates. Since different weapon systems have different maximum rates, and since the movement rate of a maneuvering unit is limited by the rate of the slowest weapon system, faster weapon systems have periods of time when they are stationary. Maneuver movement is controlled entirely by the Ground Combat Model, which determines detection capabilities, vulnerability, and weapon system capabilities.

(c) Stay Activity. The model also simulates stationary activities for all gamed units not engaged in other specified activities; i.e., all units that are not performing another military activity such as firing, moving, or combat. Whether or not addressed by gamer orders, an inactive gamed unit will consume classes I and III or IIIA supplies and can be assessed as to losses and status; other units can gain information about the inactive unit. If a unit has completed all its orders before the end of a game period, the unit will automatically stay until the end of the period. Stay activity orders can be written to command ground units to remain in position for a specified length of time or until a specified game time arrives.

(d) Engineer. The Engineer Model simulates the scheduling and execution of engineer activities associated with the construction and destruction of obstacles and facilities. The model accepts engineer tasks, assigns task priorities, determines task feasibility, mobilizes mission units to execute the tasks, simulates the engineering activity in terms of time and materiel resources used, and demobilizes the mission units.

1. Obstacles and facilities are parts of an overall barrier plan developed for the game being conducted. Engineering activities can be initiated by gamer order to start work on a specified obstacle or facility or by request from the Movement Model when some engineering activity is necessary for the conduct of a directed movement. Where the engineer activity is requested by the Movement Model, the moving unit is unable to complete its move until the engineer activity is completed.

2. Engineer task priorities are based primarily upon the urgency of the activity in terms of its impact on the force's overall plan of maneuver. Task feasibility is determined in terms of task site (proximity to FEBA) and time and materiel availability.

3. The Engineer Model automatically allocates resources to each feasible task, constructs a mission unit to execute the task, moves the mission unit to the task site, simulates the initiation of work when sufficient resources are on site, periodically updates task status until completion of the task (or until a gamer order to stop the task is encountered), and returns the mission unit to its origin.

(4) Airmobile Model. The Airmobile Model permits simulation of a variety of airmobile operations. To maintain a high degree of flexibility in application of the model, the simulation depends upon the gaming staff for most of the general planning and decision making prior to execution of an airmobile operation. These plans are relayed to the model by a set of

1 5 APR 1975

DSL orders. Activities actually simulated within the model include allocation of transport and escort aircraft to conduct the operation, staging and loading of the airmobile force, the actual airmobile movement, attrition of the airmobile column while in flight to and from the objective area, suppression of air defenses by escort aircraft, deplaning at a landing zone, refueling and rearming of aircraft, and the release of aircraft from operational control of the airmobile force.

(5) Combat Service Support Model. This model simulates the resupply of manpower and materiel to units within the DIVWAG system. The model deals with personnel replacements, replacement of major items of equipment, and resupply of critical consumables such as food (class I), fuel (class III and IIIA), barrier materials (class IV), and ammunition (class V).

(a) Replacement of personnel and major items is accomplished once for each simulated day of combat. Availability of replacement personnel and major end items is on a daily basis, input by the gamer, with available assets accumulating over time; i.e., assets not used on previous days are available in addition to those available for the current time. Requirements are based on unit losses, represented by the authorized unit level of personnel and major items less the quantities on hand within the unit at the time of the replacement action. First priority for replacements and major end items is to front line maneuver units, second priority to reserve maneuver battalions and all artillery units, and third priority to all other units. If sufficient replacements or major end items are not available to fill the needs within a unit priority group, each unit receives a pro rata share of available resources based on amounts required by all units within the priority group. Replacement and major end items arrive at the receiving units after an appropriate travel delay.

(b) The treatment of resupply of consumables within the Combat Service Support Model is conceptually similar to treatment of replacement of personnel and major items. Implementation differs to account for the following:

1. No limitation is placed on quantities of consumables available to the force. In the case of consumables, the primary limiting factor is the availability of transportation to move the materiel from various supply points to the consumer. The model treats movement of consumables through a series of supply points from the nominal point of entry into the force to the using unit. The model uses either a unit distribution or a supply point distribution method on each leg of the supply chain, depending upon the supply class of the consumable and the nature of the receiving unit at each node.

2. To accomplish a more continual flow of consumables, resupply requirements are determined and actions initiated on a more frequent basis than with replacements. The model currently uses a 2-hour cycle for all consumables except class I, which is on a once-a-day cycle. (As experience is gained with the model, some appropriate cycle between the extremes of hourly and daily requirement determination should be established.)



15 APR 1975

3. A request for resupply of consumables is generated if the quantity in the unit trains is less than a fixed percentage of the authorized amount in trains. That percentage is currently set at fifty percent.

NOTES

I-3-14

## CHAPTER 4

### COMPUTER SYSTEM REQUIREMENTS

The minimum computer system requirements for execution of the DIVWAG model are listed below.

Core storage capacity	124,000 words (octal)
Magnetic tape drives	2
Magnetic disk capacity	3,000,000 (decimal)
Printer	1
Card reader	1
Software features	FORTTRAN compiler (ANSI) <sup>1</sup> Overlay loader (2 levels) Mass storage input/output

---

1. American National Standards Institute Incorporated, American National Standard FORTRAN, ANSI X 3.9; 1966.

NOTES

I-4-2

## APPENDIX A

### DEFINITIONS OF TERMS

The terms and abbreviations herein are defined as used within the DIVWAG system and may differ from terms defined in AR 320-5, Dictionary of Army Terms; however, AR 320-5 should be used in conjunction with this appendix for a more complete definition of terms.

AAFSS: advanced aerial fire support system.

actual ground zero: the geographical location of detonation of a nuclear weapon.

AD: air defense.

ADAFSS: Army direct aerial fire support system.

ADCU: see air defense capable unit.

ADM: atomic demolition munition.

ADP: automatic data processing.

AGZ: see actual ground zero.

airbase: a base of operations for fixed and/or rotary wing aircraft from which mission aircraft or air units may be dispatched.

air burst: a burst of a nuclear weapon the fireball of which does not touch the earth's surface.

air defense capable unit: any unit possessing defined air defense weapons.

A-kill: damage inflicted upon an aircraft by defending air defense units in which the aircraft is not recoverable for future use.

AM/HI: Airmobility in the Mid/High Intensity Environment; a USACDC study.

analysis of variance: the parametric technique for testing whether several samples have come from identical populations, thus determining the significance of variations in a group of samples.

Analysis Output Processor: an element of the DIVWAG system which produces and arrays data for analysis at the termination of a given set of gaming conditions.

analysis plan: a plan for the analysis of data produced from the application of a model or simulation.

AOP: see Analysis Output Processor.

APC: armored personnel carrier.

area reconnaissance: reconnaissance of a rectangular area defined by four sets of coordinates.

ASR: see available supply rate.

available supply rate: the rate of consumption of ammunition that can be allocated considering the supplies and facilities available, for a given period.

background conditions: the visual conditions surrounding a target which affect the capability of enemy weapons to acquire it as a target.

backorder: that portion of supplies requisitioned which is not immediately available for supply but which will be recorded as a commitment for future issue; in the DIVWAG system it is the amount of a commodity which a unit orders to allow it to maintain its authorized level assuming that the commodity is consumed at a mean usage rate.

band: one of a number of equally dimensioned subdivisions integral to a simulated military unit and paralleling the front of the unit.

barrier: any object--man made or natural--which restricts, delays, or stops the movement of a force or imposes losses of personnel or equipment on units crossing it.

barrier line: a continuous series of barrier segments.

barrier segment: a unique homogeneous element of a barrier defined by its two end points and a mnemonic identification.

Bartlett statistic: a sampling statistic for Bartlett's test of homogeneity of variances.

basic unit: the smallest unit described within the DIVWAG data base by a table of organization of which the total task organization is composed. A basic unit cannot be further subdivided.

battlefield orientation: model approximation to the axis of advance described by the slope in the X-Y plane of a line connecting the centers of mass of the two forces.

begin morning nautical twilight: the time at which the rising sun is 12 degrees below the horizon.

B-kill: damage inflicted upon an aircraft requiring a forced landing of the aircraft; if in enemy territory, the aircraft is not recoverable for future use.

AD-A112 688

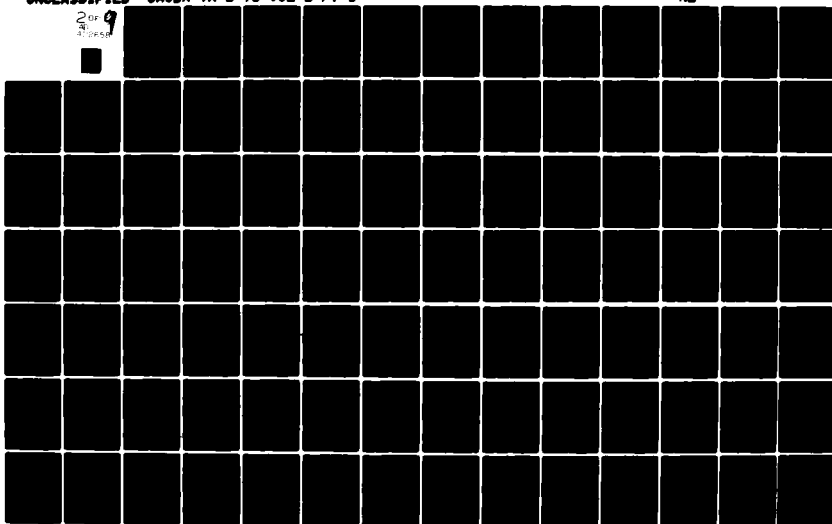
ARMY COMBINED ARMS COMBAT DEVELOPMENTS ACTIVITY FORT--ETC F/8 18/7  
DIVAS MODEL DOCUMENTATION. VOLUME II. PROGRAMMER/ANALYST MANUAL--ETC(U)  
JUL 76

UNCLASSIFIED

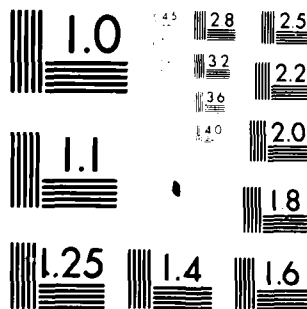
CACDA-TR-8-76-VOL-2-PT-1

NL

2 of 9  
4 of 58



11265



MICROCOPY RESOLUTION TEST CHART  
NBS 1963-A



blowdown: the effect of static overpressure or dynamic pressure from nuclear blast resulting in felled man made objects or trees.

BMNT: see begin morning nautical twilight.

CAS: see close air support.

CEP: see circular error probable.

CFS: see constant fire subsegment.

chi-square: a measure of the discrepancy existing between observed and expected frequencies.

circular error probable: the radius of a circle within which half of the projectiles are expected to fall.

C-kill: damage inflicted upon an aircraft in which the aircraft is forced to discontinue its mission because of damage to the aircraft or because of a pilot or copilot casualty.

class of supply: the subdivision of commodities necessary to equip, maintain, and operate a military command, generally so divided for planning and administration of the supplies. For definitions of classes of supply see Chapter 9, FM 100-10.

close air support: air action against hostile ground targets that are in close proximity to friendly ground forces and that require detailed integration of each air mission with the fire and movement of those forces.

closed game: a game in which the control team and opposing player teams use separate rooms or areas, maintain separate maps and records, and in which a controlled amount of intelligence is given to each player team.

CM/CB: see countermortar/counterbattery.

collection system: an integrated system of individual sensor types or groups of sensor types.

combat service support: the assistance of providing administrative services, chaplain services, civil affairs, finance, legal service, maintenance, medical service, military police, replacements, supply, transportation, and other logistical services. Within the DIVWAG system, it includes personnel replacement, supply, and some transportation.

combat support: operational assistance furnished combat elements by other designated units.

complex unit: a unit comprised of two or more basic units.

conditional casualties: casualties which would be incurred if a weapon were firing alone and not receiving return fire.

Constant Data Input Processor: the element of the DIVWAG system which contains the group of special purpose data load programs that read source data from cards, convert the data to the form required by the Period Processor, and load the data onto DIVWAG data files.

constant fire subsegment: flight segment within which an air unit is expected to be subject to attrition by a unique and constant set of air defense weapons.

consumable: item that is normally expended and used up beyond recovery in the use for which it was designed or intended.

cookie cutter: a slang expression used to describe a circular radius of effects for indirect fire weapons, including nuclear weapons; contrasted with precise calculation of effects considering the influence of terrain and other factors.

countermortar/counterbattery: relating to the detection, suppression, or destruction of enemy indirect fire by mortars or artillery.

CSS: see combat service support.

cycle time: the duration of battle which is processed by a single pass through the Ground Combat Model.

DAF: see direct aerial fire.

damage radius: radius of an equivalent circular area damaged by a nuclear blast.

default rate: rate used when no rate is specified.

delayed casualties: casualties for which the manifestation of the cause is delayed over time; e.g., nuclear radiation.

delivery lead time: time required for a unit to obtain supplies after the order has been placed.

desired ground zero: the geographical location at which munition detonation is desired.

deterministic: not subject to random variation.

DGZ: see desired ground zero.

diagnostic message: a statement printed by a computer program indicating a possible or known erroneous condition detectable by the program.

direct aerial fire: close air support delivered by Army aircraft in close proximity to friendly forces.

direct support: a condition whereby a force is required to support another specific force and is authorized to answer directly the supported force's request for assistance.

division force: a military organization of division size and designation supported by appropriately assigned or attached combat, combat support, and combat service support forces; a modern and more definitive term similar to division slice.

DIVTAG II: a short title for Division Through Army Group Model (second iteration).

DIVWAG: Division War Game; a short title used in lieu of the term Division War Game Model.

DIVWAG Scenario Language: the special purpose English-like language used by the gaming group to direct simulated units within the DIVWAG system to carry out desired activities.

D-kill: damage inflicted upon an aircraft in which the aircraft is able to continue its mission.

DLT: see delivery lead time.

DMF: see dominant mask function.

dominant mask function: a function describing terrain masking in terms of distance to and height of terrain features masking the line of sight of a specific geographical location.

dose rate: the amount of radiation received by personnel per unit of time.

DPFO: data processing field office.

driver, driver routine: a routine within a computer program the principal function of which is to call other routines at the required times and perform related control functions.

DS: see direct support.

DSL: see DIVWAG Scenario Language.

DSL conditional: a statement included in a DSL order which describes the conditions under which the order is to be executed. Also, a DSL statement describing the conditions under which a battle is to be terminated.

DSL modifier: a qualifying statement included in a DSL order which specifies the precise activity desired.

DSL order: a complete DSL description of an activity to be performed by a designated unit including the type of activity and associated DSL modifiers.

ECM: see electronic countermeasures.

EENT: see end evening nautical twilight.

effectiveness indicator: selected quantitative elements of the large body of performance data available, chosen to highlight the most significant aspects of the performance data in light of the force evaluation objectives and the particular measures of effectiveness they support.

EIC: see equipment item code.

electronic countermeasures: those measures taken to eliminate or reduce the effectiveness of equipment or units relying upon electronic emanations for the performance of an assigned mission(s).

elevation grid: an arbitrary grid imposed upon a geographical area the points of which have elevations specified in the data base.

end evening nautical twilight: the time at which the setting sun is 12 degrees below the horizon.

EOH: see equipment on hand.

equipment item code: a numerical code (1 to 200) assigned to unique items of equipment and supplies, including munitions, simulated within the DIVWAG system.

equipment on hand: materiel physically in the possession of a designated military organization, installation, or activity and in a useable status and condition.

error analysis: the procedure of locating the cause of erroneous results produced by the model or associated computer programs.

error message: same as diagnostic message.

event sequence: the ordering of events in chronological sequence.

expendable: same as consumable.

facility: a physical entity on the battlefield which facilitates the movement of military forces; e.g., bridge, ferry, ford, bypass; a military installation at which specified activity takes place.

FARMWAG: Field Army Modernization War Game.

FARR: see forward area refuel and rearm.

FEBA: see forward edge of battle area.

fire cycle time: the time required by a weapon/ammunition type to acquire a target and to aim, fire, and deliver one round within the Ground Combat Model.

fire mission: specific assignment to a fire unit. In the DIVWAG system, this term refers to the specific set of data describing a single target communicated to the TACFIRE Model.

fire support coordination center: the location at which are centralized communications facilities and personnel incident to the coordination of all forms of fire support. In the DIVWAG system this term applies to that portion of the Intelligence and Control Model regulating supporting aerial fire and artillery.

flight corridor: a rectangular area centered along the flight path of an aircraft or group of aircraft.

flight leg: a segment of a flight path defined by two end points.

forest type index: an index between zero and three designating the extent of forest cover in a given terrain cell.

forward area refuel and rearm: relating to a highly mobile supply point in the forward area of the battlefield used to refuel and rearm attack, escort, or lift aircraft.

forward edge of battle area: foremost limits of a series of areas in which ground combat units are deployed excluding covering or screening forces. This edge is represented within the DIVWAG system by a straight line perpendicular to the battlefield orientation and equidistant from the forward maneuver elements of each force.

free game: a game conducted without benefit of a fixed set of rules.

Friedman test: nonparametric test based on ranks that can be used when matched subjects are obtained. Friedman's two-way analysis of variance by ranks provides a test of the null hypothesis that k related samples were drawn from k identically distributed populations.

FSCC: see fire support coordination center.

function of land combat: one of five recognized activities of land combat: firepower; mobility; intelligence; logistics; command, control, and communications.

game: simulation of a situation of competition or conflict in which the opposing players decide which courses of action to follow on the basis of their knowledge about their own situation and intentions and on their information about their opponent's courses of action.

game cycle: the procedures to be followed in the execution of a game period to maintain the gaming rate.

game period: an arbitrary time interval for the execution of game activities which is uninterrupted by game direction.

game period concept: a document prepared by the control group and approved by the game director stating the military and system considerations and guidelines governing activities of each game period.

game period turnaround: the process that occurs from the end of one set of game period computer runs until the beginning of the next set of game period computer runs.

game plan: the written concept by which the game manager plans to accomplish the game objectives.

gaming rate: the amount of simulated combat time that can be processed per unit of real time expended by the gaming staff.

GCM: Ground Combat Model.

general support: that support given to the supported forces as a whole and not to any particular subdivision thereof.

general support reinforcing: that support provided by an artillery unit with the mission of supporting the forces as a whole and of providing reinforcing fires for another artillery unit.

glimpse rate: the number of discrete instances at which a sensor is active and able to acquire targets per unit of time.

ground sensor terminal: a ground-based facility with equipment for receiving directly and processing information acquired by airborne sensor systems.

ground zero: a point on the earth's surface at or immediately below the detonation of a weapon.

GS: see general support.

GST: see ground sensor terminal.

GZ: see ground zero.

height of burst: the height of a munition detonation measured from the earth's surface.

HOB: see height of burst.

homoscedastic: having equal variances.

hypothesis: an assumption or statement about the probability distributions of the population.

impact radius: the radius around an aim point within which indirect fire munitions may be expected to impact.

IMPWAG: Improvement of the ICAS War Game Model; a USACDC study.

INC: Intelligence and Control; a short title used to identify the Intelligence and Control Model.

induced radiation barrier: a barrier to movement created by induced radiation from a nuclear burst.

induced radiation hazard area: an area in which measurable induced radiation from a nuclear burst is present and in which the level of radiation is hazardous to personnel.

in-flight attrition: the destruction of aircraft while the aircraft is in aerial flight.

initial supply point: the origin for materiel and supplies within the combat zone.

intelligence processing: the activity which converts information of the enemy, weather, and terrain into finished intelligence for use by a commander in the decision-making process.

intercept segment: same as constant fire subsegment.

intermediate supply point: a point in the chain of supply between the initial supply point and either a division distributing point or a recipient unit.

IR: infrared.

IS: see intercept segment.

iteration, iteration time: the duration of unit-pair engagement which is processed by a single pass through the target acquisition, firepower distribution, and firepower effectiveness submodels of the Ground Combat Model.

Kolmogorov-Smirnov statistic: a statistic used to test the null hypothesis of normality of the sample distribution.

Kruskal-Wallis test: a nonparametric test based on ranks.

landing zone: the geographical point of debarkation of an airmobile force from its airlift transportation.

level of resolution: the degree to which an entity is broken into discrete, recognizable constituent parts or elements. High resolution implies many parts; low resolution implies few parts.

level of significance: for every null hypothesis  $H_0$ , there exists at least one alternative hypothesis  $H_1$ . An a priori procedure is to reject  $H_0$  in favor of  $H_1$  if a statistical test yields a value whose associated probability of occurrence under  $H_0$  is equal to or less than some small probability symbolized as  $\alpha$ . That small probability,  $\alpha$ , is called the level of significance.

lift force: the military force which supplies ground or air transportation to another force or unit.

line of sight: unobstructed vision between two points.

LOH: light observation helicopter.

LOS: see line of sight.

major end items: a final combination of end products, component parts, and/or materials that is ready for its intended use. Used primarily to describe vehicles, weapon systems, and sensor systems.

major engagement segment: same as possible engagement segment.

Mann-Whitney U-test: a method of employing the Mann-Whitney U-statistic to make orthogonal comparisons among the k treatment population or to determine which comparisons among the k treatment population are significant.

mean use rate: the usage rate determined by calculating a statistical average of usage rates since the start of the game.

measure of effectiveness: a quantitative value that indicates the degree to which a military unit or system performs its mission or achieves its goal.

MES: see major engagement segment.

mission unit: a unit created automatically within the model for performing a model generated mission and not under direct gamer control.

mobility category: a group of unit types having similar mobility characteristics.

mobility class: a group of equipment items having similar mobility characteristics.

MOE: see measure of effectiveness.

move segment: that portion of a movement path described by a straight line and entirely within a single terrain cell.



moving target indicator: a device that limits the display of radar information primarily to moving targets.

MTI: see moving target indicator.

NAM: Nuclear Assessment Model.

nap-of-the-earth: as close to the earth's surface as vegetation and obstacles will permit flight. The model assumes a height of 50 feet above the ground.

NATO: North Atlantic Treaty Organization.

nonparametric statistics: statistical procedures that do not depend on a knowledge of population distributions and associated parameters.

nonresolution unit: a constituent of a resolution unit or a higher echelon unit; a unit without specific location and which cannot accept orders for the execution of a function of land combat or cannot directly incur losses from enemy forces.

null hypothesis: the hypothesis that there is no difference between the procedures or populations.

open game: a military game in which opposing players have complete access to information pertaining to an opponent.

order and shipping time: the period of time represented by the combined activities of ordering and delivery of supplies or materiel.

Orders Input Processor: the element of the DIVWAG system which provides the communication link from the gaming group to the Period Processor by means of DSL orders and operating instructions.

period: an arbitrary time interval for the execution of game activities which is uninterrupted by game direction.

period history tape: a magnetic tape containing data that are a representation of all significant activities simulated by a computer-assisted war game model during a game period.

Period Output Processor: an element of the DIVWAG system that produces at the completion of a game period a set of post-period reports to be analyzed and used for play of subsequent periods of a game.

Period Processor: the central element of the DIVWAG system that contains the mathematical models, data maintenance routines, and event scheduling and execution logic required to simulate the military activities portrayed in the DIVWAG Model.

pick up zone: the geographical point of embarkation of an airmobile force.

pinpointing: detection of evidence that a potential target has fired.

possible engagement segment: that continuous portion of a flight leg in which an air unit is vulnerable to fire from a unique set of air defense capable units (ADCUs); the set including at least one ADCU.

primary equipment item: an item of equipment that is not a component of a larger equipment item.

prompt casualties: casualties incurred by a unit at the time of a nuclear blast.

R: an empirical value used as an environment fitting parameter generally equated to mean range for line of sight in homogeneous terrain.

radiological barrier: a barrier resulting from radiation effects produced by a nuclear detonation.

reinforcing fires: fire delivered on call from the reinforced unit by a unit assigned to a reinforcing mission.

reorder cycle time: time interval between successive supply orders.

resolution unit: a military unit with a specific location which can be given specific orders for the execution of one or more functions of land combat and can incur losses from hostile fire.

rigid game: a war game conducted in strict conformance with a fixed set of rules.

roughness and vegetation index: a numerical index to a combination of terrain slope variation and nonforest vegetation.

route reconnaissance: aerial reconnaissance conducted along a designated route or line on the earth's surface.

routine: a discrete element of a computer program that performs a desired function or operation when called by another routine.

RV index: see roughness and vegetation index.

safe point: a point at which aerial vehicles are beyond the range of enemy air defense capable units; usually a point in friendly territory.

safety level: quantity of materiel, in addition to the operating level of supply, required to be on hand to permit continuous operation in the event of minor interruption of normal replenishment or unpredictable fluctuations in demand.

scenario: a narrative description of the setting in which the strategic military, political, economic, and social environment is established and the physical geography is set forth. A chronological listing of preplanned occurrences.

secondary equipment item: an item of equipment that is associated directly with, or a component of and is secondary to, a primary equipment item.

semiopen game: a military game in which opposing players have access to limited but corresponding type information of opposing forces.

sensor: a device capable of detecting the presence and/or location of possible military targets.

side analysis: analysis of factors not designated as the principal area of interest or concern.

side looking airborne radar: radar system directed perpendicular to the flight path of the transporting aircraft.

simulation: the representation of physical systems and phenomena by models or other logical or mathematical representation; a technique used to study and analyze the operation and behavior of man/machine systems in terms of the elements of which they are composed.

sky/ground ratio: ratio of the brightness of the horizon to the brightness of the target facing the target.

slant range: the distance between two points not at the same elevation.

SLAR: see side looking airborne radar.

slew rate: the rate of angular movement of a weapon or sensor in the horizontal and/or vertical plane.

sortie: the execution of an assigned mission by one aircraft.

staging area: the geographical area in which a military unit is assembled and prepared to execute an assigned operation; may be the area from which a military operation is initiated.

STANO: surveillance, target acquisition, and night observation.

stochastic: subject to random variations.

subordinate list: the listing of units that comprise and which are subordinate in command to a designated military unit.

supply point distribution: the method of distributing supplies in which the receiving unit is issued supplies at a supply point and moves the supplies to its own area using its own transportation.

surface burst: a burst of a nuclear weapon in which the fireball of the weapon touches the earth's surface.

table of organization and equipment: a table which prescribes the normal mission, organizational structure, and personnel and equipment authorization for a military unit.

TACAIR: high performance aircraft support of ground forces usually conducted by other Service or Allied air components.

TACFIRE: tactical fire direction.

target subunit: a subdivision of a potential target unit on an area basis which is considered individually for detection.

supply point: a military activity which receives, stores, and distributes supplies and equipment.

task organization: the organization of a military unit for the performance of a specific task.

terrain cell: an arbitrary geographical subdivision of the earth's surface for purposes of describing assumed homogeneous terrain characteristics.

terrain masking: the interposition of a terrain feature so as to prohibit line of sight between an observer and a candidate area or target.

TOE: see table of organization and equipment.

trafficability index: a numerical index that describes the capability of a portion of the earth's surface (soil) to sustain military operations.

transfer order: a DSL order directing a change to the task organization.

travel mode: a four-character alphabetic code describing the type of movement a unit is to perform.

UGS: see unattended ground sensor.

UID: see unit identification.

unattended ground sensor: a ground sensor which is unattended by human operators.

unit distribution: that method of distributing supplies in which all requested items are transported from the supply point to the requesting unit using vehicles provided by the supplying unit.

unit identification: an eight-character alphanumeric designation of a military unit, installation, or activity for identification purposes.

unit type: a categorization of units having identical tables of organization and equipment.

unit type designator: a four-character alphanumeric designation for a unit type.

update cycle: the interval between successive recalculations of battlefield geometry parameters.

UTD: see unit type designator.

volley: a fire mission consisting of one round fired by each weapon of a unit.

WAGCAP: Improvement of the War Game Capability; a USACDC study; a related follow-on study is designated WAGCAP II.

war game: a simulation, by whatever means, of a military operation involving two or more opposing forces, conducted using rules, data, and procedures designed to depict an actual or assumed real life situation.

weapon munition index: a four-character alphanumeric designator for a unique combination of weapon type and ammunition type.

weather zone: a rectangular area of game play over which weather conditions are considered uniform.

zero period: a short game period processed prior to initiation of the planned scenario for the purpose of model initialization.

NOTES

I-A-16

SECTION II

CONSTANT DATA INPUT PROCESSOR





## CHAPTER 1

### GENERAL DESCRIPTION OF THE CONSTANT DATA INPUT PROCESSOR

1. INTRODUCTION. The purpose of this section is to identify the constant data requirements of the DIVWAG system and provide instructions for entering the data into the constant data files. Constant input data is that information in the DIVWAG system that need not be changed throughout the conduct of the dynamic play of a war game.
2. SIGNIFICANCE OF THIS SECTION. The conduct of a war game using the DIVWAG system requires the utilization of a large data base containing many thousands of data items. These data must be derived or extracted from available sources, coded, and entered into storage devices. This section provides: (1) a description of the data required as input by DIVWAG prior to the actual conduct of a game to produce data for evaluation and detailed instructions for entering the data onto standardized card formats compatible with the programs that actually load the data from cards into the constant data files, (2) a complete program description including input and output variables to each routine, a logical flow, and a flow chart, (3) an output description identifying all printed output reports, and (4) a source listing of each load program.
3. ORGANIZATION OF THIS SECTION. Each of the subsequent 16 chapters of this section pertain to a specific data load process of the Constant Data Input Processor. The purpose of each is briefly described in the following subparagraphs.
  - a. Executive control. The Executive Control routine (PREDIV) provides the ability to execute any number of constant data input processors within one job request, thus providing an overall increase in efficiency to the DIVWAG system.
  - b. Tables of Organization and Equipment. The Tables of Organization and Equipment (TOE) are generally the first to be entered into the DIVWAG system constant data as common information for all models. The information needed to satisfy TOE data requirements includes the force structure, senior and subordinate units, and primary and secondary items of equipment. TOE sets the stage for all other data entries for the DIVWAG system.
  - c. Task Organization. The fourth chapter of this section deals with task organization input data. These data establish the identification and composition of units to be used in the game, the task force organization or organizational structure of the units within a force which will be used at the start of the game, the initial location of units, initial supporting roles of appropriate units, and the basic logistical network of the force.
  - d. Environment. Chapter 5 of this section deals with weather and terrain descriptions. This is common file information, used by several of the DIVWAG models. Preparation of this type data is time consuming and should be initiated as one of the first tasks of a war game effort.

e. Unit Geometry. Chapter 6 of this section deals with unit dimensions and item distribution data. This is data file 28 information and is used by several of the DIVWAG models.

f. Intelligence and Control. The seventh chapter is one of the most complex of the entire section. Data requirements are varied, and the complexity of preparation and entry exceeds that for the other models. For the most part the Intelligence and Control Model uses its own unique data. In the process of accessing its unique data bases, it provides an output that is used by virtually all other models.

g. Ground Combat. In Chapter 8 the direct fire weapons are addressed, and their constant data requirements are defined. The sensors for target detection and recognition are described with their performance data and the weapons to which they feed information. These weapons are matched with their ammunition, and target defeat is described in terms of hit probabilities and kill probabilities.

h. Area Fire Assessment. The entry of the constant data for accessing the loss and destruction created by conventional indirect fires is the type of data entered in this model. Chapter 9 discusses personnel protection against hostile fires through their posture on the battlefield, shielding by combat equipment (tanks), and areas of lethality.

i. TACFIRE. In the tenth chapter is decision type information that an artillery personnel will recognize as needed for division artillery and its firing batteries. Target priorities must be established, with the types of weapons and ammunition to be used; and the method of attack must be determined.

j. Air Ground Engagement. Chapter 14 is concerned with data to describe close air support provided by Army attack helicopters and Air Force aircraft. The data cover 40 type close air support missions and include various mission descriptors, in terms of numbers and types of aircraft required, and mission results. The data also include air defense and aircraft parameters required to treat aircraft attrition en route to and from the target area.

k. Suppression. Within the DIVWAG system suppression is treated as the interruption of an activity in response to incoming fires. Chapter 12 treats the data required, generally the duration of such interruptions.

l. Nuclear Assessment. The Nuclear Assessment Model deals with firing, detonation, and results of nuclear devices. Required data are prescribed in Chapter 13.

m. Movement. In virtually any application of the DIVWAG system, there must be extensive ground movement of military units. Chapter 14 treats the entry of constant data needed for this movement. Administrative aerial movement is also treated with this data input.

n. Engineer. The Engineer Model treats the construction and destruction of obstacles to and facilities for ground movement. The instructions for entering data required in this model are found in Chapter 15.

o. Airmobile. Chapter 16 deals with data unique to the Airmobile Model. This model, which treats the airmobile movement of units, also uses data associated with several other models, particularly Air Ground Engagement, Combat Service Support, and Movement.

p. Combat Service Support. The resupply of friendly troops either from supply points or through unit distribution is one of the major concerns of this model. Chapter 17 provides the instructions for constant data entry to satisfy model demands.

4. SEQUENCE OF LOAD PROGRAMS. Insofar as is possible, each set of data described in Chapters 3 through 17 is independent of other data to the extent that the sequence in which the data are loaded is not critical. Exceptions are listed below, with a suggested sequence of loading.

a. Data Areas Depending on Previous Loads:

(1) Task Organization. The task organization data (Chapter 4) cannot be loaded until after Organization and Equipment data (Chapter 3) and Combat Service Support Model data (Chapter 17) have been loaded. If either Organization and Equipment or Combat Service Support Model data are changed after the task organization has been loaded, then task organization must be reloaded with the new Organization and Equipment and/or Combat Service Support Model data.

(2) Movement. The Movement Model data (Chapter 14) cannot be loaded until after TOE data (Chapter 3) are loaded. If TOE data are changed after the Movement Model data have been loaded, then the Movement Model data must be reloaded.

(3) Suppression. Suppression Model data (Chapter 12) cannot be loaded until after Organization and Equipment data (Chapter 3) are loaded. Any change to Organization and Equipment data will necessitate reloading Suppression Model data.

(4) Nuclear Assessment. The data used by the Nuclear Assessment Model (Chapter 13) cannot be loaded until after conventional Area Fire Model data (Chapter 9) are loaded. Revisions to Area Fire Model data necessitate reloading Nuclear Assessment Model data.

b. Suggested Sequence of Loading. Organization and Equipment data must be loaded early in the data preparation cycle. This is reinforced by the fact that practically every data file is keyed in some sense to the Organization and Equipment data, either through equipment item codes or unit type designations. The following sequence of loading is suggested, based both upon the program requirements listed above and upon situations where knowledge of what has been loaded in one area is critical to the logical definition of data in other areas.

(1) Organization and Equipment data (Chapter 3) should be the first set of data loaded. Knowledge of these data is needed to prepare virtually all other data programs.

(2) Environment data collection (Chapter 5) should be initiated early because of the volume of data required.

(3) Intelligence and Control Model data input (Chapter 7), while not prerequisite to other data, is the most complex package and should be initiated early in the cycle.

(4) Combat Service Support Model data preparation (Chapter 17) is relatively straightforward but should be stabilized early because it is prerequisite to the task organization.

(5) Air Ground Engagement Model data requirements (Chapter 11) are voluminous, and preparation should be initiated early in the data development process. The individuals developing Air Ground Engagement Model data should ideally develop the Airmobile Model data (Chapter 16) thereafter because of a strong interrelation between data requirements of the two models. These individuals should be familiar with both models prior to working on either one, since heavy use of Air Ground Engagement Model data is made by the Airmobile Model.

(6) The Area Fire Model (Chapter 9), TACFIRE Model (Chapter 10), and Nuclear Assessment Model (Chapter 13) data requirements are strongly interrelated and should, if possible, be developed by the same individuals.

(7) Task organization data (Chapter 4) should be entered into the system at approximately this stage of development. Individuals preparing the task organization data should be those who developed the Organization and Equipment data and are familiar with its content.

(8) Engineer Model data (Chapter 15) may be prepared late in the data development cycle. Planning for this load, however, should be initiated early if extensive barrier plans are to be used.

(9) Ground Combat Model data requirements (Chapter 8), are voluminous but are relatively independent of other data areas.

(10) Movement Model (Chapter 14) and Suppression Model (Chapter 12) data may enter the system late in the cycle. Proper preparation of Movement Model data requires knowledge of the model's interface with barriers. The Suppression Model data requirements are simple and may be fulfilled last in the data preparation cycle.

## CHAPTER 2

### EXECUTIVE CONTROL DATA INPUT

The routine PREDIV reduces the number of cards and separate computer runs required to load the constant data and increases the overall efficiency of the DIVWAG system. This routine, by making all the load programs accessible on disk, allows the programs to be executable without a compile and allows as few as one or all constant data decks to be loaded with one job request. A softfall capability of PREDIV allows any data set to be skipped if the data are incorrect for loading and to continue to the next data set. If any data set is skipped, a message will be printed informing the programmer that a data set has not loaded.

NOTES

II-2-2

## APPENDIX A

### INPUT REQUIREMENTS FOR EXECUTIVE CONTROL DATA INPUT

The sample job control deck assembly is shown in Figure II-2-A-1, and a sample constant data input control deck is shown in Figure II-2-A-2. Each set of constant data is separated by a card identifying the data load program; i.e., GCMLD....PREP is the Ground Combat Model load program resident on disk. The last card behind all of the data decks is a STOP....PREP card. Only one stop card is required for one computer run. The cards required for each of the constant data decks are shown in Figure II-2-A-1. When preparing the cards, the program name begins in card column 1 and PREP must be entered in card columns 77-80 on each card. The appropriate constant data deck will follow the corresponding control deck card as shown in Figure II-2-A-2.

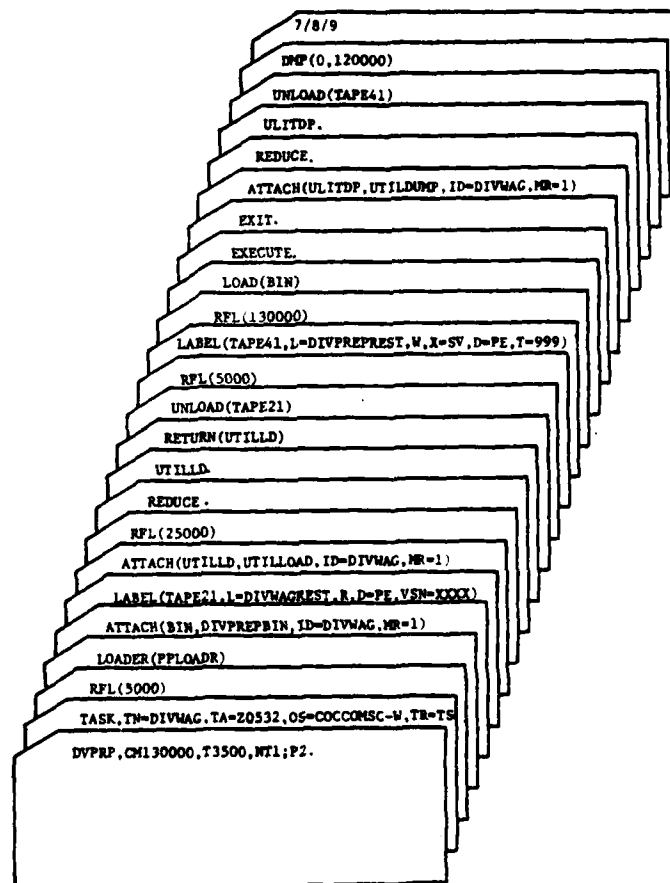


Figure II-2-A-1. Executive Job Control Deck Assembly



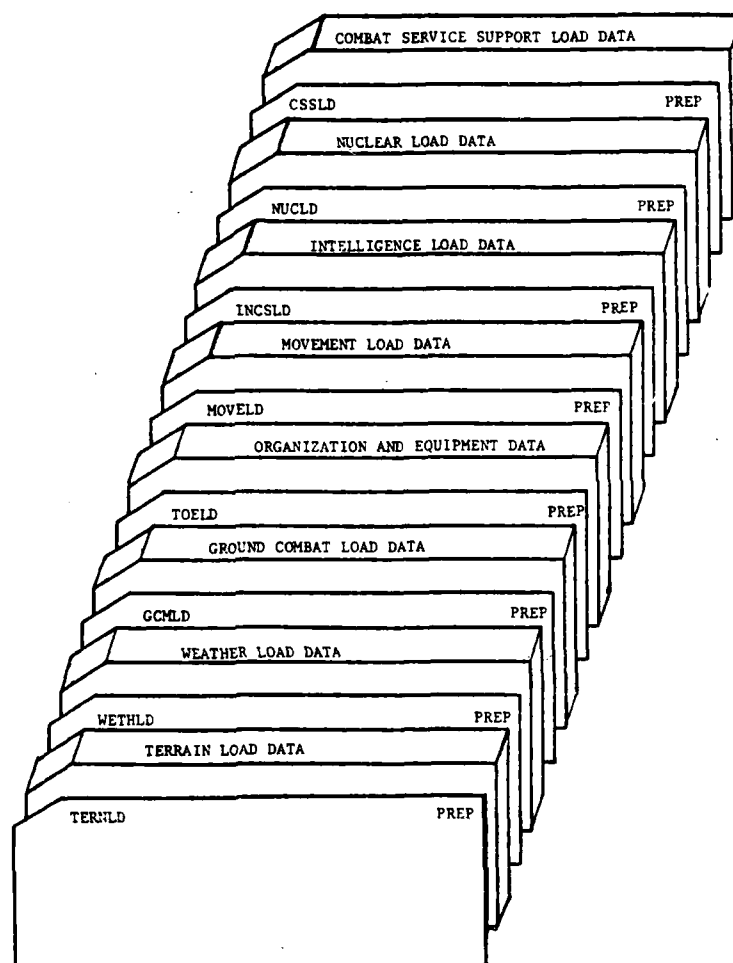


Figure II-2-A-2. Executive Control Constant Data  
Control Deck Assembly  
(Continued on Next Page)

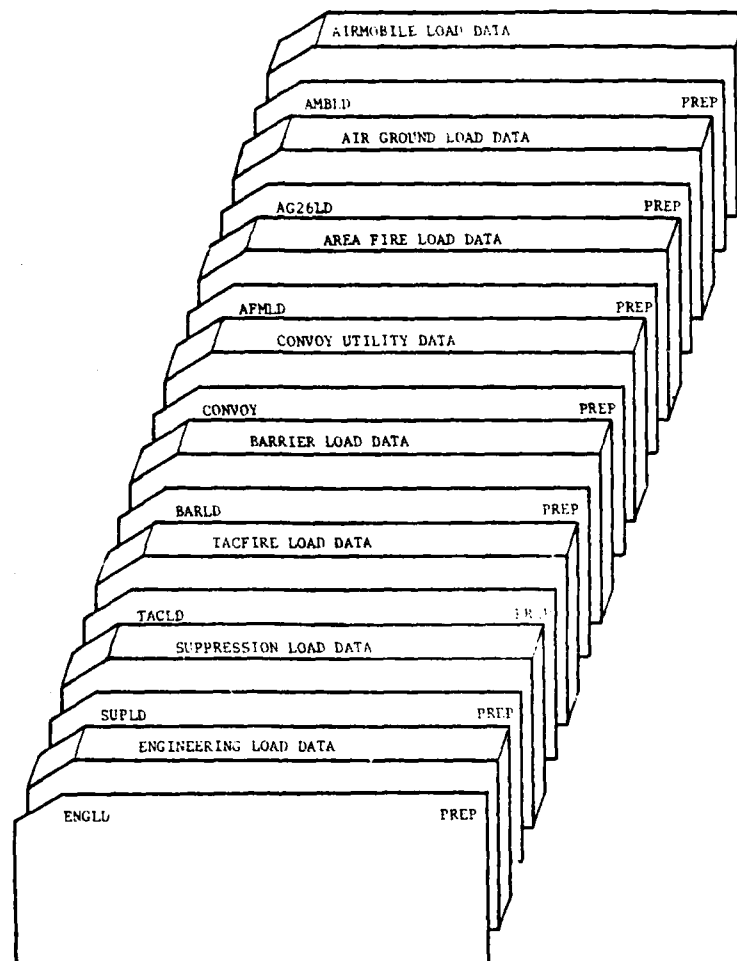


Figure II-2-A-2. Executive Control Constellation Data Control Deck Assembly (Continued)

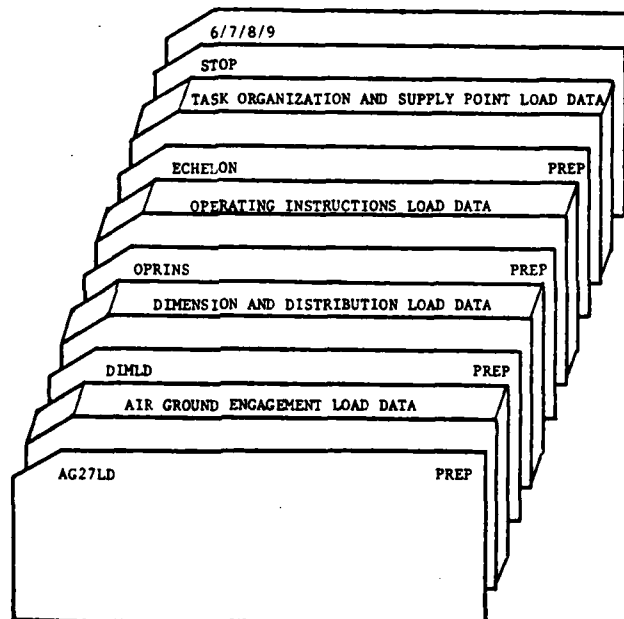


Figure II-2-A-2. Executive Control Constant  
Data Control Deck Assembly  
(Concluded)

NOTES

II-2-A-6

## APPENDIX B

### EXECUTIVE CONTROL DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. This appendix contains the program description of routine PREDIV, the primary executive control routine of the Constant Data Input Processor.

2. ROUTINE PREDIV:

a. Purpose. Routine PREDIV calls the proper overlays as needed.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IFNT(56,3)	Disk	File name table.
LOADTP	Card	Variable that identifies the constant data input routine to call.

c. Output Variables: The file name table (IFNT) is printed initially and after each overlay call.

d. Logical Flow (Figure II-2-B-1):

(1) Blocks 1 and L10. A call is made to routine GETFLE to retrieve the file name table and the table is printed.

(2) Blocks L15 and 2. A data card is read. If the card format is incorrect, another card is read.

(3) Block 3. If the last card read was a stop card, transfer control to block L400.

(4) Block 4. If this is an Engineer (ENGLD), Suppression (SUPLD), TACFIRE (TACLD), or barrier (BARLD) load card, transfer control to block L70.

(5) Block 5. If the first character of the requested load is greater than M, transfer control to block L3.

(6) Block 6. If the first character of the requested load is greater than G, transfer control to block L2.

(7) Block 7. If the first character of the requested load is greater than C, transfer control to block L1.

(8) Blocks 8 and 9. If the requested load is a Combat Service Support load (LDCSS) set the overlay flag to six, and transfer control to block L200.

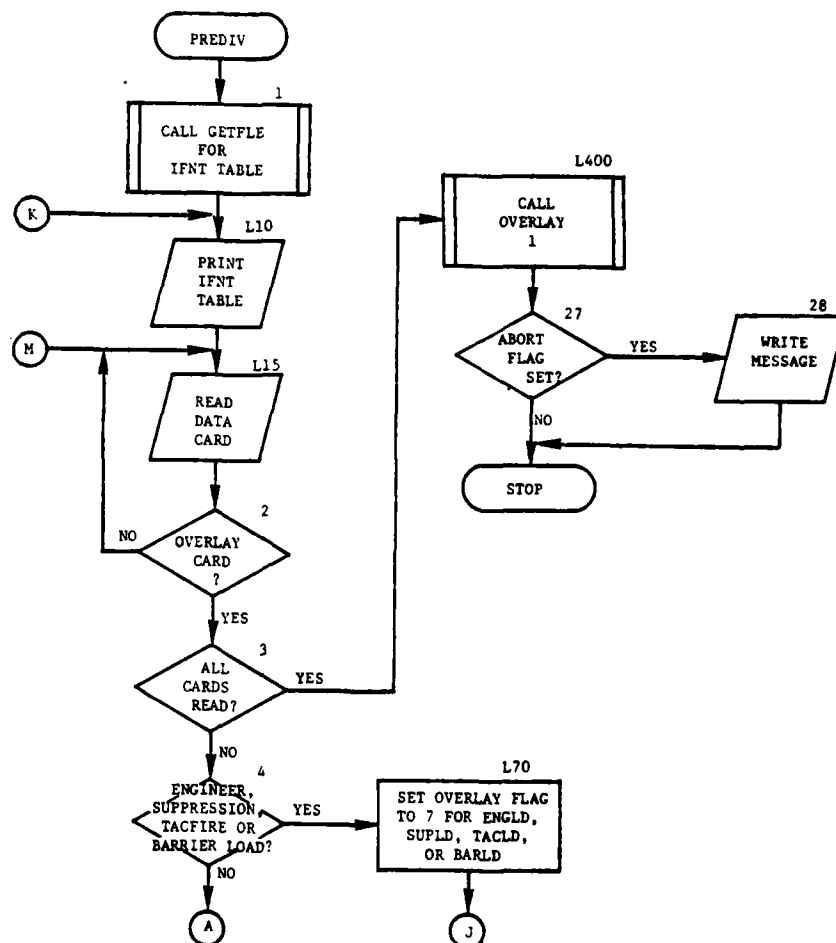


Figure II-2-B-1. Routine PREDIV (Continued on Next Page)

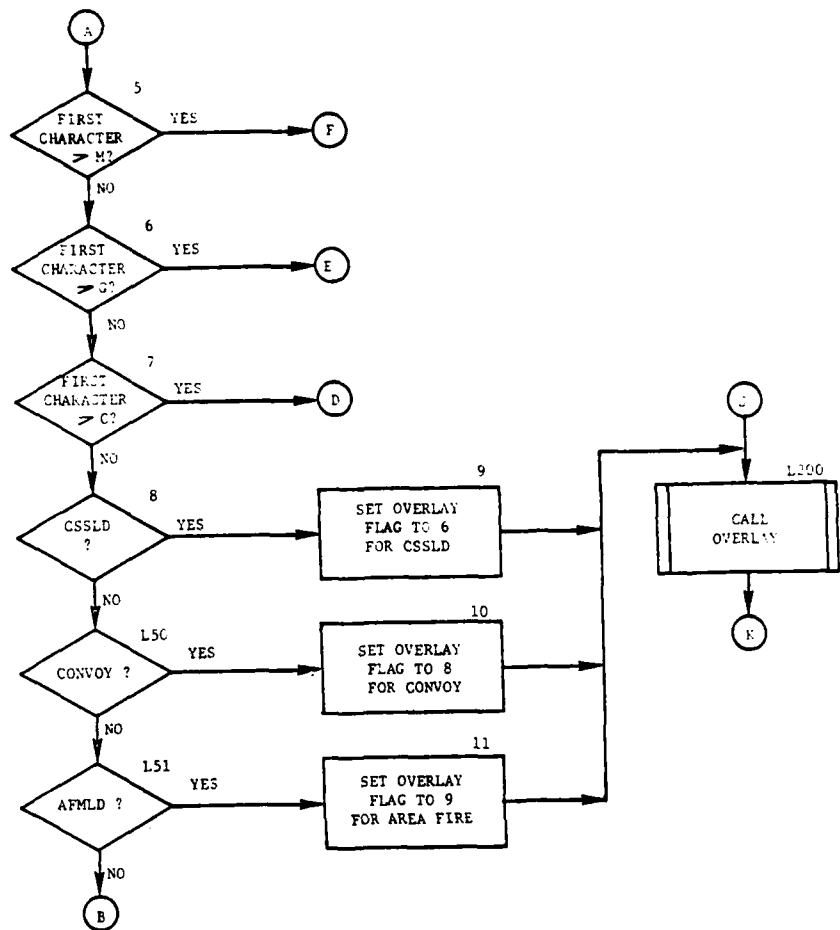


Figure II-2-B-1. Routine PREDIV (Continued)

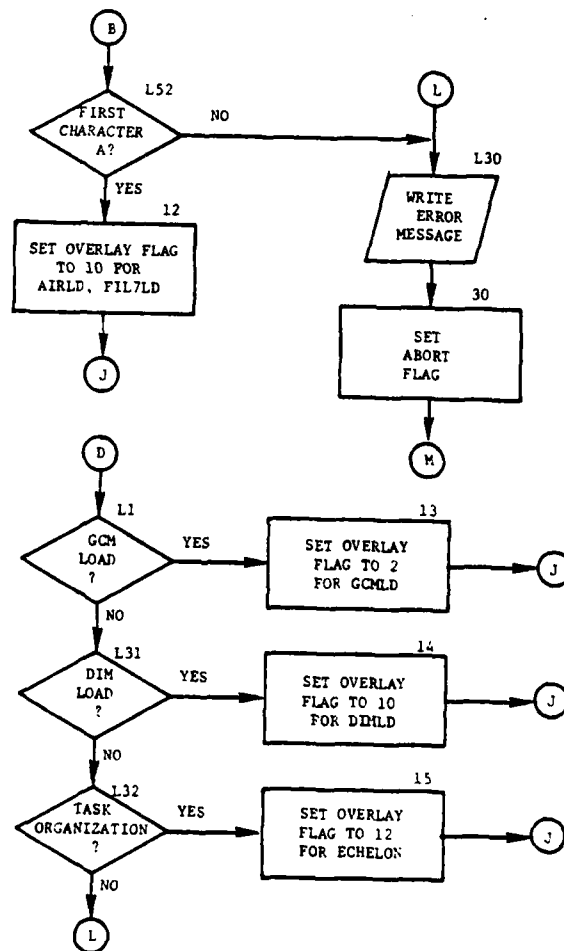


Figure II-2-B-1. Routine PREDIV (Continued)



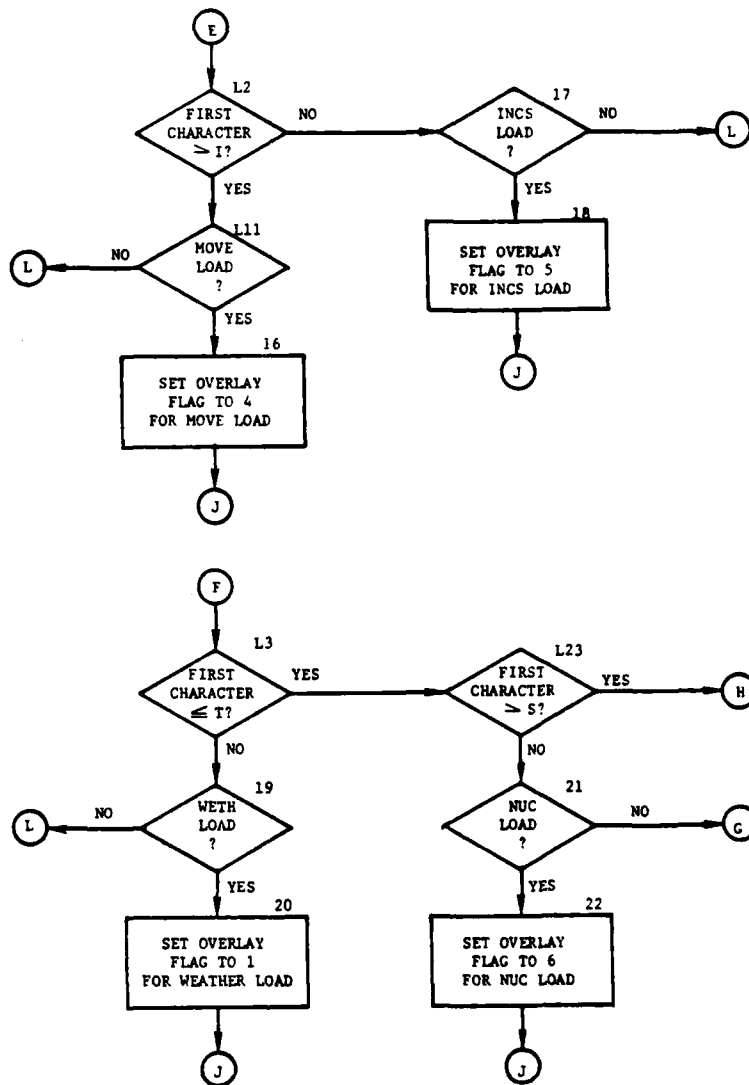


Figure II-2-B-1. Routine PREDIV (Continued)

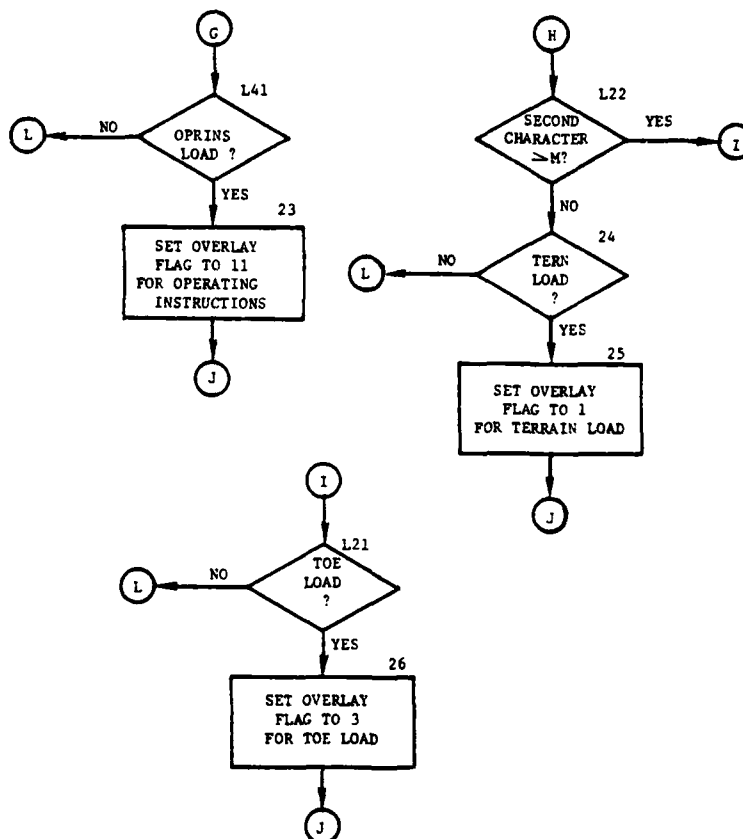


Figure II-2-B-1. Routine PREDIV (Concluded)

- (9) Blocks L50 and 10. If the requested load is convoy (CONVOY), set the overlay flag to eight, and transfer control to block L200.
- (10) Blocks L51 and 11. If the requested load is Area Fire (AFMLD), set the overlay flag to nine, and transfer control to block L200.
- (11) Blocks L52 and 12. If the first character of the overlay type is A, set the overlay flag to 10, and transfer control to block L200; otherwise, transfer control to block L30.
- (12) Block L70. Set the overlay flag to seven, and transfer control to block L200.
- (13) Blocks L1 and 13. If the requested load is Ground Combat (GCMLD), set the overlay flag to two, and transfer control to block L200.
- (14) Blocks L31 and 14. If the requested load is unit dimensions (DIMLD), set the overlay flag to 10, and transfer control to block L200.
- (15) Blocks L32 and 15. If the requested load is task organization (ECHLD), set the overlay flag to 12, and transfer control to block L200.
- (16) Block L2. If the first character of the load type is less than or equal to I, transfer control to block 17.
- (17) Blocks L11 and 16. If the requested load type is movement (MOVLD), set the overlay flag to four, and transfer control to block L200; otherwise, transfer control to block L30.
- (18) Blocks 17 and 18. If the requested load is Intelligence and Control (INCLD) set the overlay flag to five, and transfer control to block L200; otherwise, transfer control to block L30.
- (19) Block L3. If the first character of the load type is less than or equal to T, transfer control to block L23.
- (20) Blocks 19 and 20. If the requested load is weather (WETLD), set the overlay flag to one, and transfer control to block L200; otherwise, transfer control to block L30.
- (21) Block L23. If the first character of the load type is greater than S, transfer control to block L22.
- (22) Blocks 21 and 22. If the requested load is Nuclear Assessment (NUCLD), set the overlay flag to six and transfer control to block L200.
- (23) Blocks L41 and 23. If the requested job is operating instructions (OPRINS), set the overlay flag to 11, and transfer control to block L200; otherwise, transfer control to block L30.
- (24) Block L22. If the second character of the load type is greater than M, transfer control to block L21.

(25) Blocks 24 and 25. If the requested load is terrain (TERLD), set the overlay flag to one, and transfer control to block L200; otherwise, transfer control to block L30.

(26) Blocks L21 and 26. If the requested load organization and equipment (TOELD), set the overlay flag to three; otherwise, transfer control to block L30.

(27) Block L200. Call the system overlay routine to load the requested overlay; then, transfer control to block L10.

(28) Block L400. Call the system overlay routine to overlay segment 1, This segment executes a dump.

(29) Blocks 27 and 28. If the abort flag is set, write an informative message and stop the program.

(30) Blocks L30 and 30. Write an error message, set the abort flag, and transfer control to block L15.

3. ROUTINE OVRLY7: Routine OVRLY7 determines which secondary overlay (ENGLD, SUPLD, TACLD, or BARLD) is requested and loads that segment.

4. ROUTINE OVLY10: Routine OVLY10 determines which secondary overlay (AIRLD, FIL7LD, or DIMLD) is requested and loads that segment.

## APPENDIX C

### OUTPUT DESCRIPTIONS FOR EXECUTIVE CONTROL DATA INPUT

The Constant Data Input Processor Executive Control routine will print the file name table (IFNT) before and after each requested constant data load program is executed.

NOTES

II-2-C-2

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR EXECUTIVE CONTROL

(AVAILABLE UNDER SEPARATE COVER)

II-2-D-1

NOTES

II-2-D-2



## APPENDIX A

### ORGANIZATION AND EQUIPMENT DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for organization and equipment are documented in Appendix A, Unit Representation Input Requirements, to Chapter 3 of the Period Processor (Section IV).

NOTES

II-3-A-2

## APPENDIX B

### ORGANIZATION AND EQUIPMENT DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. Loading of tables of organization and equipment (TOE) data is accomplished by routine LDTOE which, in turn, calls routine DMPTOE to provide a listing of the data. LDTOE is supported by utility routines SORT and SRCH.

2. ROUTINE LDTOE:

a. Purpose. This routine loads secondary equipment tables, data file 6; unit type designator (UTD) directory, data file 51; basic unit strength tables, data file 52; and UTD breakdown tables, data file 53.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ICARD	Card	Data cards defining senior/subordinate units, TOE, secondary equipment list, and bulk-loaded expendable supplies.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
IVEC6	DF6	Secondary equipment quantity pairs.
IVEC53	DF53	Breakdown of complex UTDs.
ITOE	DF52	Contains quantities for respective equipment items on hand and bulk in trains.
IVEC51	DF51	UTD directory. The first 250 entries contain ordinals for data file 52, the last 750 entries contain ordinals for data file 53.

d. Logical Flow (Figure II-3-B-1):

(1) Block 1. Call routine GETFLE to retrieve the file name table (IFNT) so that the Input/Output package may function.

(2) Block 2. If the dump flag is on, control goes to block L700.

(3) Block 3. Call routines REMOVE and CREATE for each of the files being created in this routine, which initializes these files on the DIVWAG data file.

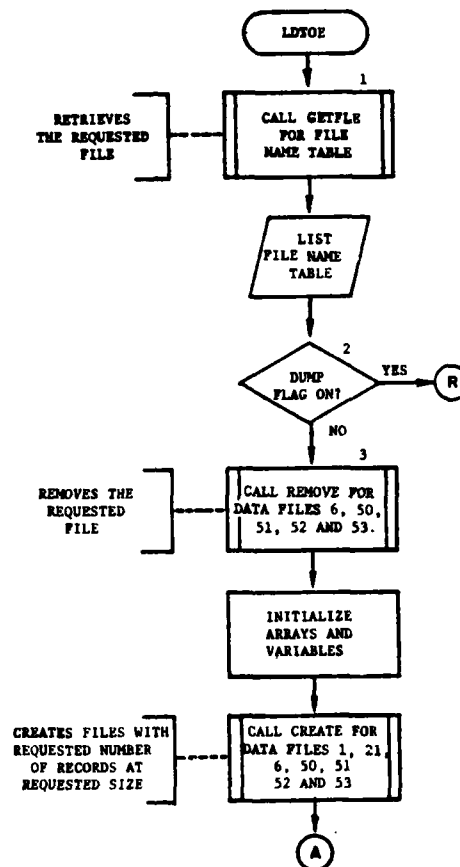


Figure II-3-B-1. Routine LDTOE. (Continued on Next Page)

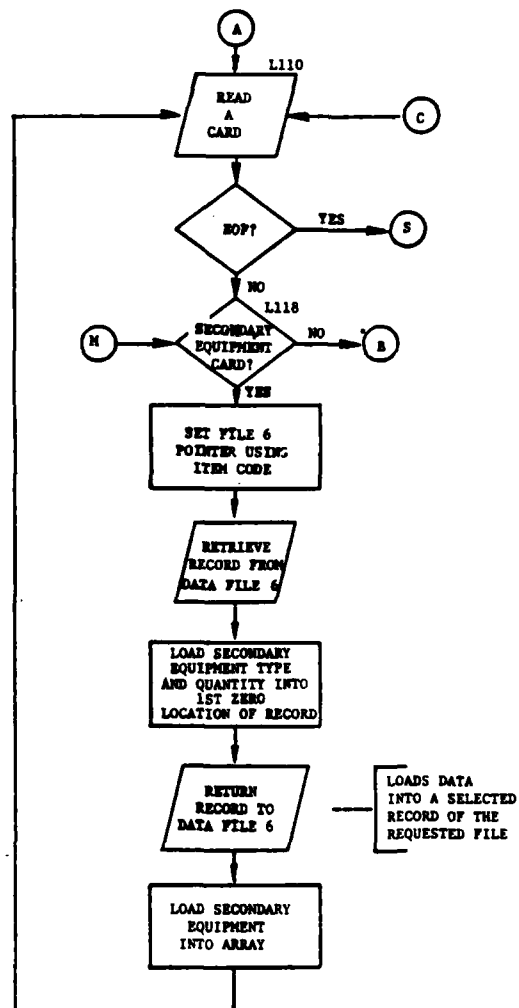


Figure II-3-B-1. Routine LDT0E. (Continued)

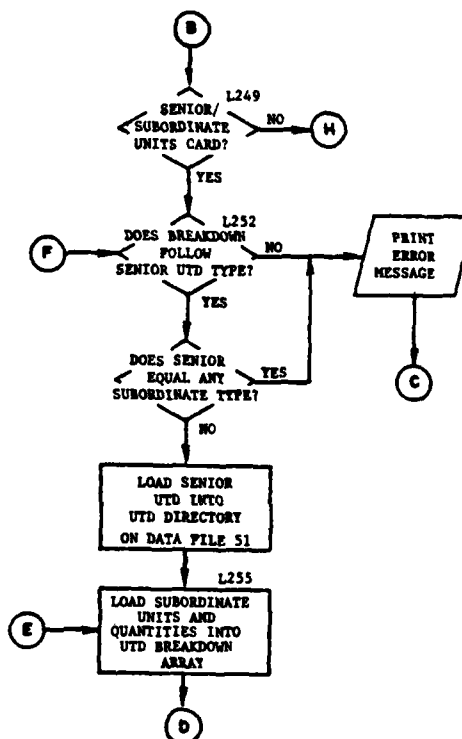


Figure II-3-B-1. Routine LDTOE. (Continued)

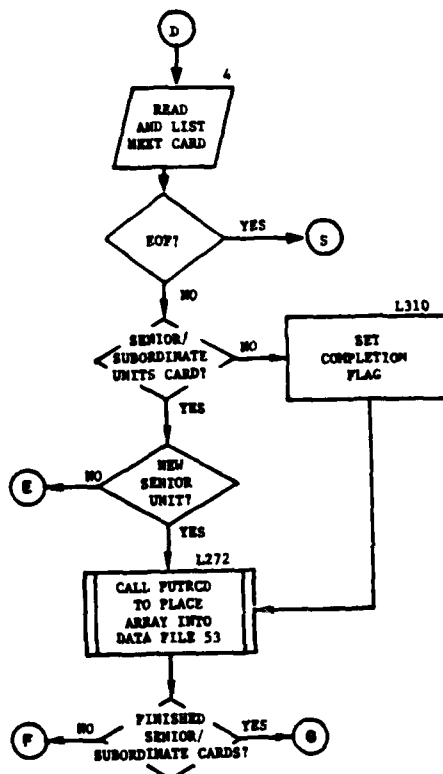


Figure II-B-3-1. Routine LDT0E. (Continued)





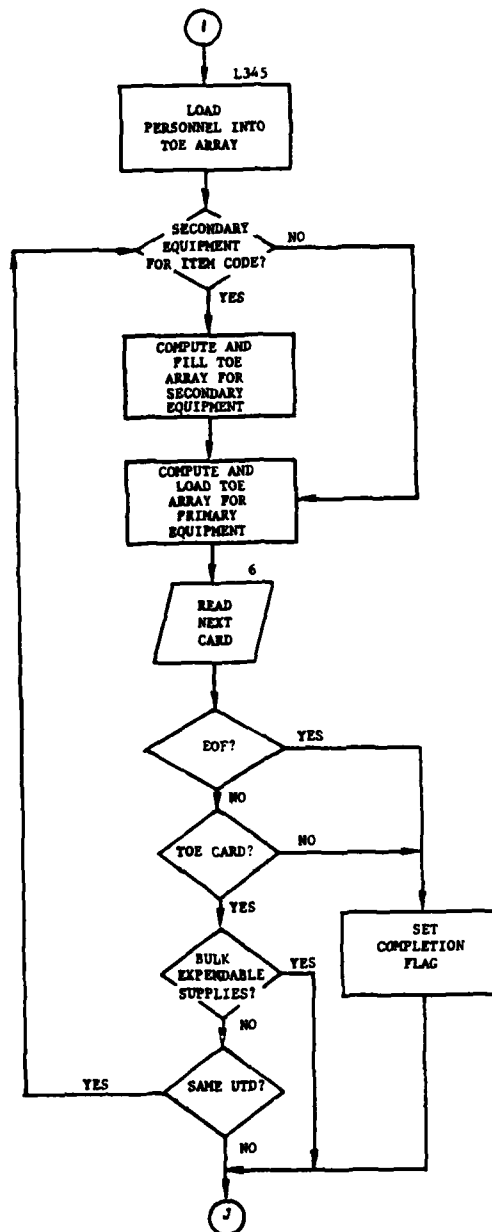


Figure II-3-B-1. Routine LDTOE. (Continued)

II-3-B-7

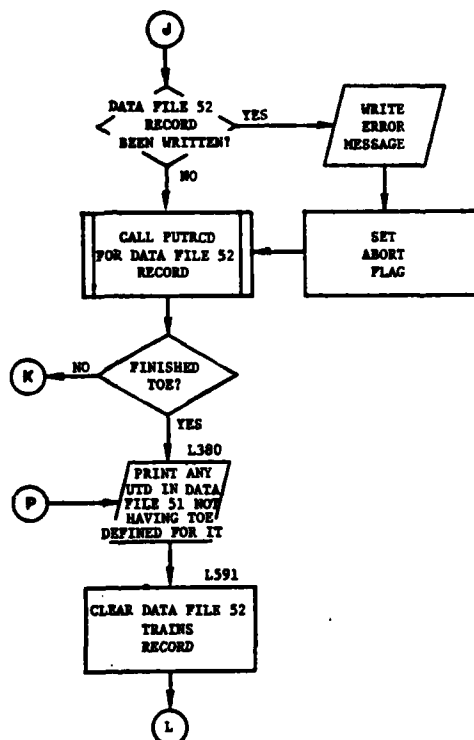


Figure II-3-B-1. Routine LDTOE. (Continued)

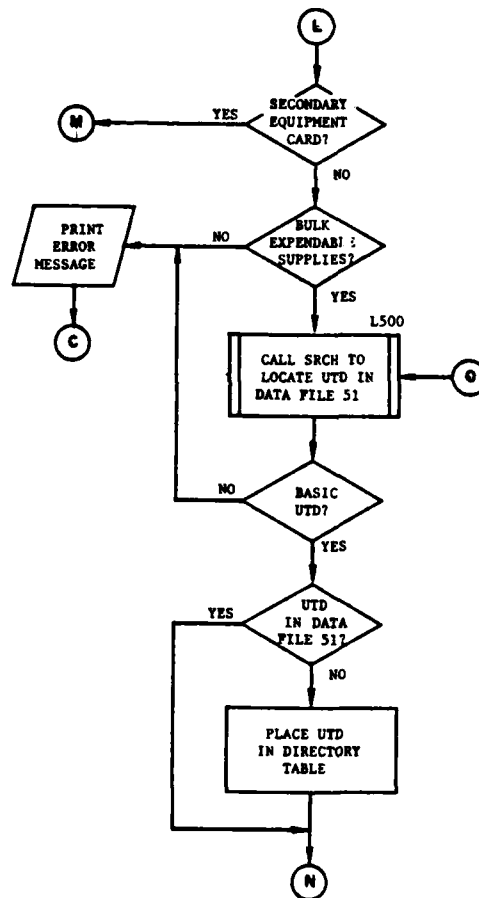


Figure II-3-B-1. Routine LDTOE. (Continued)

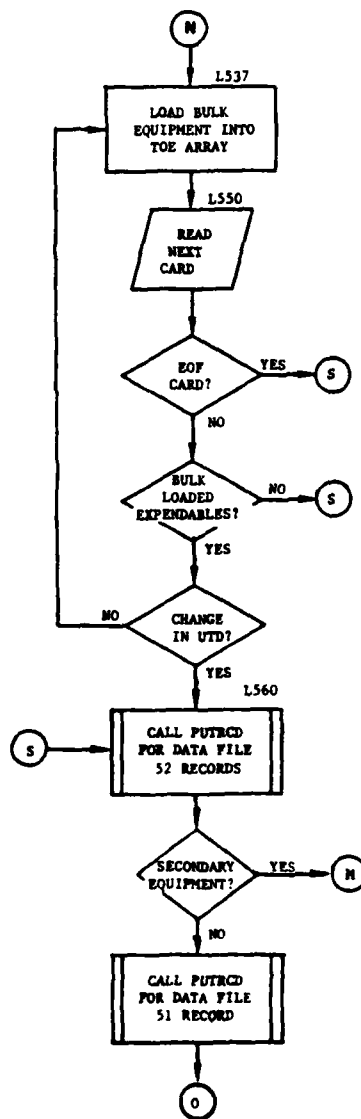


Figure II-3-B-1. Routine LDTOE. (Continued)

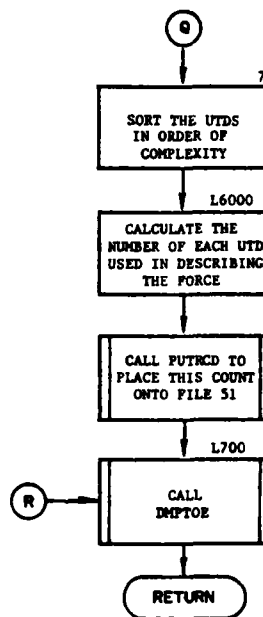


Figure II-3-B-1. Routine LDTOE. (Concluded)

(4) Block L110. A data card is read. If an end of file was sensed, control transfers to block L560. The card image is printed.

(5) Block L118. A check is made to determine if the card is a secondary equipment card. If not, control transfers to block L249; otherwise, routine GETRCD is called to retrieve the appropriate data file 6 record. The card data are loaded into this record starting at the first zero location, and routine PUTRCD is called to return the updated record to the file. The data file 6 record is then loaded into the secondary equipment array for later use, and control returns to block L110.

(6) Block L249. The card previously read is not of the secondary equipment type. If it is of TOE or bulk-loaded expendable supplies type control goes to block L325.

(7) Block L252. The card previously read is of the senior/subordinate type. The array for holding data file 53 records is initialized. A check is made to determine if the senior UTD was previously loaded into the complex portion of data file 51. If it was, an error message is printed and control returns to block L110; if it was not, it is loaded into the data file 51 array.

(8) Block L255. The subordinate UTDs and quantities are loaded into the data file 53 array.

(9) Block 4. The next data card is read, and if an end of file was sensed control transfers to block L560. If the card is not of the senior/subordinate UTD type control transfers to block L310; if the card is of senior/subordinate type and the senior UTD did not change, control returns to block L255.

(10) Block L272. The data file 53 record counter is incremented and routine PUTRCD is called to put the data file 53 array into data file 53. If all senior/subordinate cards have not been processed control returns to block L252.

(11) Block 5. Each data file 53 record is retrieved and each UTD of each record is examined. Each UTD, not of a complex type, is loaded into the basic portion of data file 51 until it is filled.

(12) Block L310. The completion flag is set and control transfers to block L272.

(13) Block L325. If the card is not of TOE or bulk-loaded expendable supplies type, control transfers to block L560. If it is bulk-loaded expendable supplies type control goes to block L380.

(14) Block L325. Routine SRCH is called to locate the unit type designator (UTD) in the UTD directory table. If it is a complex UTD, an error message is written and control returns to block L110. If it was not found, in the table, place it in the UTD directory table.

(15) Block L345. This block updates the TOE array for data file 52. For each equipment item code of the TOE, data file 6 is searched for secondary equipment. If none is found the TOE entry for that primary item is updated by the appropriate quantity. If secondary equipment is defined for a primary item code a nesting effect may occur, as each secondary equipment item may in turn have secondary equipment items. The products of these items are carried through into the TOE sums.

(16) Block 6. The next data card is read. If an end of file was sensed or the card is not TOE the completion flag is set. If the UTD does not change control returns to block L345. If the data file 52 record of this UTD has not previously been written, routine PUTLCD is called to place the TOE data on data file 52. If TOE data processing is not completed control returns to block L335.

(17) Block L380. Check to determine if all UTD entries in data file 52 have TOEs defined. If not, an informative message is written.

(18) Block L591. Initialize trains area of data file 52. If this is a secondary equipment type card control goes to block L118. If this is a bulk expendable supplies type card control goes to block L500; otherwise, an error message is printed and control returns to block L110.

(19) Block L500. Routine SRCH is called to locate the UTD in data file 51. If the UTD is of a complex type, an error message is printed and control returns to block L110. If the UTD was located control goes to block L537; otherwise, the data file 52 record counter is incremented and the UTD is placed in data file 51.

(20) Block L537. The bulk quantities are loaded in their appropriate equipment item code positions in the TOE array.

(21) Block L550. The next card is read. If it is an end of file card (9999), an end of file was sensed, or it is not a bulk-loaded expendables card; control goes to block L560. If the UTD did not change control returns to block L537.

(22) Block L560. Routine PUTLCD is called to place the bulk expendables records onto data file 52. If the last card is secondary equipment type control returns to block L118. The UTD directory is placed onto data file 51.

(23) Block 7. The list of complex UTDs for each force is arranged so that the components of each UTD are either basic or less complex UTDs. The resulting order is stored in the ORDER array. The first UTD associated with the Blue force must be AABB and the corresponding Red force UTD is AARR. By definition these UTDs are also the most complex UTDs associated with the respective force. They are placed last in the ORDER array. The ordering of the remaining UTDs is accomplished by attempting to match each component of the next UTD in the list with entries in the list of basic UTDs or with complex UTDs having components already resolved. If no match is found for a component of this UTD it is placed next to the last in the ORDER array,

the other UTDs are moved up one, and the next UTD is processed. If UTDs are discovered containing unresolved components, an error message is written and processing of this force's UTDs is terminated.

(23) Block L6000. This area calculates the total number of UTDs of each type that is used in describing the force. It is accomplished by working from the bottom of the ORDER array derived in block 7. The running total for each UTD is kept in the KNT51 array. There can be only one AABB or AARR UTD in each force, so a one is placed in that position in KNT51. The remaining UTDs are counted by considering each UTD in the order of complexity (from the bottom of the ORDER array). The product of the number of times it has been used and the amount of each component UTD it requires is added to the value associated with each component UTD in KNT51. KNT51 is then placed as the second record of data file 51.

(24) Block L700. Routine DMPTOE is called.

3. ROUTINE SRCH:

a. Purpose. This routine finds the ordinal number of the passed UTD in the UTD directory table.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IREC	Call	UTD
IVEC51	Call	UTD directory table.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
IREC	Call	Ordinal of UTD.

d. Logical Flow (Figure II-3-B-2):

(1) Block 1. If the passed UTD is blank, control transfers to block L15.

(2) Block 2. A search is made through the UTD directory table for a matching UTD. If one is found, control transfers to block L20.

(3) Block L15. The UTD passed was either blank or was not found in the UTD directory table. The ordinal is set to zero and control returns to LDTOE.

(4) Block L20. The ordinal is set to the location of the UTD in the UTD table and control returns to LDTOE.



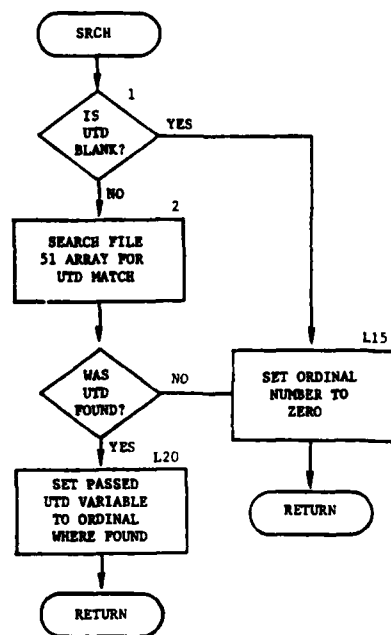


Figure II-3-B-2. Routine SRCH

II-3-B-15

#### 4. ROUTINE DMPTOE

a. Purpose. The purpose of routine DMPTOE is to dump the UTD directory table, the UTD breakdown table, the basic unit TOE table, and the secondary equipment table.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IUTD	DF51	List of UTDs.
KVEC6	DF6	Secondary equipment type and quantities for Red and Blue forces.
IVEC	DF53	UTD breakdown table. A breakdown of complex UTDs by type and amount.
KVEC	DF51	Maximum number of UTDs by type per force.
JVEC	DF52	Amount of equipment authorized on vehicles by type.
IXVEC	DF52	Amount of equipment authorized in trains by type.

c. Output Variables. A listing of the variables as noted under input variables.

d. Logical Flow (Figure II-3-B-3):

(1) Block 1. Routine GETRCD is called to retrieve the first record of data file 51. This is the UTD directory table of which the first 250 entries comprise basic UTDs, with ordinals pointing to data file 53 for UTD breakdown.

(2) Block 2. Routine GETFLE is called to retrieve data file 6, the secondary equipment file. There are 400 records, 10 words per record. The first 200 records are for the Blue force and the last 200 are for the Red force. Each record contains equipment type and amount for five items.

(3) Block 3. The secondary equipment types and amounts are listed for Blue and Red forces.

(4) Block L4. For each entry in the complex UTD table GETRCD is called to retrieve the corresponding data file 53 record. This provides a breakdown of the complex UTDs. This record is then listed.

(5) Block L20. Routine GETRCD is called to retrieve the second record of data file 51, which contains the maximum number of UTDs by type per force.

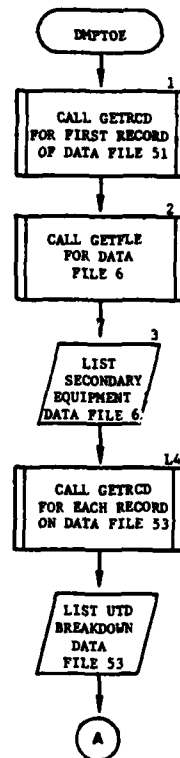


Figure II-3-B-3. Routine DMPTOE. (Continued on Next Page)

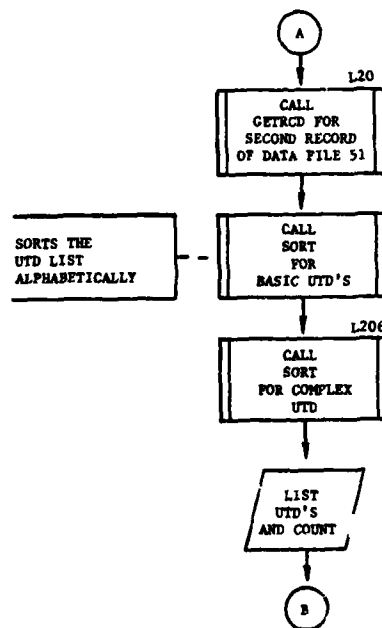


Figure II-3-B-3. Routine DMPTOE. (Continued)

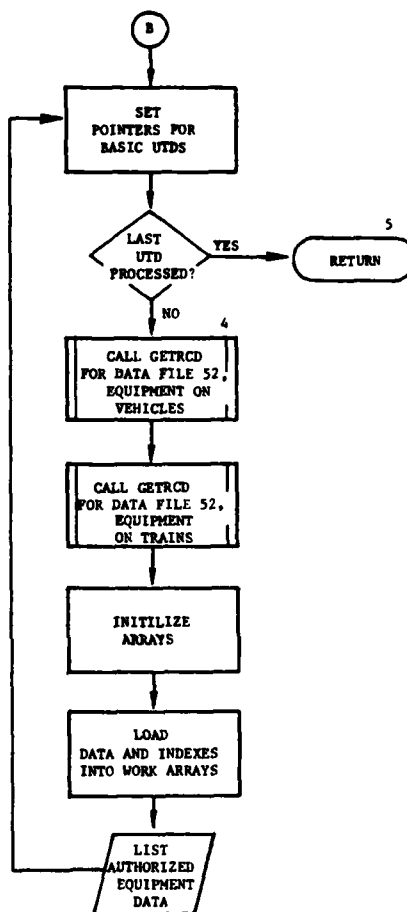


Figure II-3-B-3. Routine DMPTOE. (Concluded)

(6) Block L206. Routine SORT is called for the basic and complex units to sort the UTDs alphabetically. The UTDs are listed in sorted order.

(7) Block 4. For each basic UTD the corresponding equipment authorized in vehicles and on trains records are retrieved from data file 52. For each record, equipment types with the amounts are loaded into intermediate arrays and listed.

(8) Block 5. After data have been listed control returns to the calling routine.

#### 5. ROUTINE SORT:

a. Purpose. The purpose of routine SORT is to sort the UTD table alphabetically and arrange the table index in the same order.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
N	Call	Number of entries in the UTD table.
IA	Call	UTD table.
IB	Call	Maximum number of UTD types allowable per force.
INDEX	Call	Ordinal of each UTD entry.

c. Output Variables. The above input variables in sorted order.

d. Logical Flow (Figure II-3-B-4):

(1) Block 1. The scan limit of the sort is set to one less than the number of UTD's to be sorted.

(2) Block 2. A do loop used to scan through the UTD array is established. The value of this do loop index is the ordinal of the left UTD to be compared.

(3) Block 3. The initial ordinal of the right UTD to be compared is established.

(4) Block L3. If the value of the ordinal of the right UTD is greater than the number of entries in the UTD array control goes to block L20.

(5) Block 4. The UTDs of the left and right ordinal are compared. If the left UTD is less than the right UTD control goes to block L15.

(6) Block L11. The left and right UTD's and their indexes are interchanged.

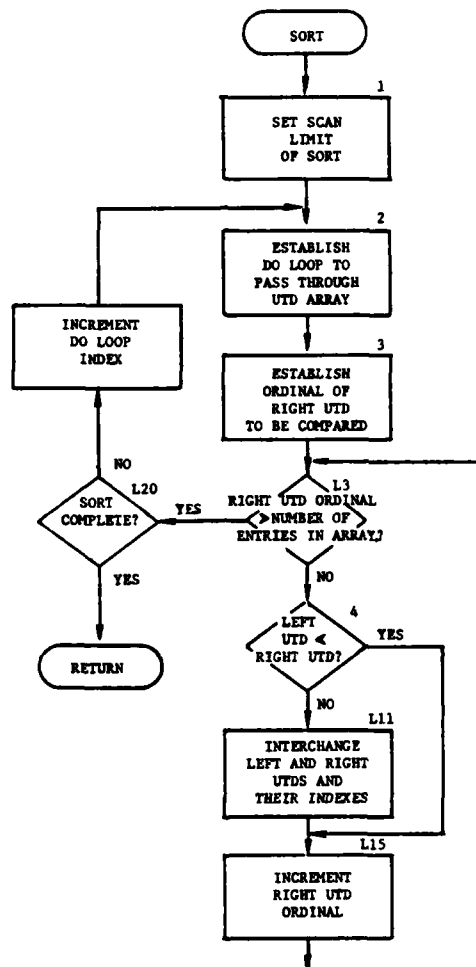


Figure II-3-B-4. Routine SORT

(7) Block L15. The ordinal of the right UTD is incremented and control returns to block L3.

(8) Block L20. If the sort is complete (the do loop ordinal would be greater than the scan limit established in block 1) transfer control to the calling routine; otherwise, the do loop index is incremented and control is returned to block 2.



## APPENDIX C

### ORGANIZATION AND EQUIPMENT DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. Five printouts are produced when the constant data input cards are read into the DIVWAG System. These five printouts are listed in Figure II-3-C-1, TOE Type Printouts. The printouts are used to validate the accuracy of the information entered as the constant data input. The format of each type printout is illustrated in the following pages together with a brief explanation of its contents.

Title of Printout	Source of Data
80-Column Card Image of Input Data	Cards in data deck
Secondary Equipment for Blue and Red	Data file 6, secondary weapons
Unit Type Designator Breakdown Table Red and Blue	Data file 53, UTD breakout
Unit Type Designator Directory	Data file 51, UTD directory
Authorized Equipment Strengths	Data file 52, authorized strength type basic unit

Figure II-3-C-1. TOE Type Printouts

2. EIGHTY-COLUMN CARD IMAGE OF INPUT DATA. The source of this input printout form is the original cards that brought the data into the DIVWAG System. Figure II-3-C-2 illustrates the type printout. At the far right is listed the number 5201, which is the card identifier. On that same line at the far left is the number 2, which indicates that this is card type 2. Figure II-3-C-2 shows card image TOE for bulk-loaded-expendables data. The headings of the card columns from the card format defines the individual data items. For a complete description of the input card image, refer to Appendix A of Chapter 3, Unit Representation Input Requirements, of the Period Processor.

3. SECONDARY EQUIPMENT FOR BLUE AND RED. Figure II-3-C-3 Secondary Equipment for Blue, illustrates the format of this constant data input printout. This is the first of the printouts having columnar titles and formatted for easy reading. The data portrayed in the printout were entered in card format, processed, and entered into data file 6. The printout

**Figure II-3-C-2. 80-Column Card Image**

PRIMARY		SECONDARY EQUIPMENT FOR BLUE									
		SECONDARY					SECONDARY				
EOH	AMT.	EOH	AMT.	EOH	AMT.	EOH	AMT.	EOH	AMT.	EOH	AMT.
21	21	0	0	0	0	0	0	0	0	0	0
22	1000	0	0	0	0	0	0	0	0	0	0
23	137	27	114	20	1	0	0	0	0	0	0
24	80	0	0	0	0	0	0	0	0	0	0
25	137	30	63	20	1	0	0	0	0	0	0
26	5	0	0	0	0	0	0	0	0	0	0
27	36	36	6	0	0	0	0	0	0	0	0
28	116	20	1	0	0	0	0	0	0	0	0
29	375	38	15	39	1995	36	10	0	0	0	0
30	150	20	1	0	0	0	0	0	0	0	0
31	40	0	0	0	0	0	0	0	0	0	0
32	135	53	10	54	50	20	0	0	0	0	0
33	11	66	17	0	0	0	0	0	0	0	0
34	300	57	0	0	0	0	0	0	0	0	0
35	300	59	2	0	0	0	0	0	0	0	0
36	64	61	1	0	0	0	0	0	0	0	0
37	190	72	0	0	0	0	0	0	0	0	0
38	105	74	0	0	0	0	0	0	0	0	0
39	155	9	1800	0	0	0	0	0	0	0	0
40	59	0	0	0	0	0	0	0	0	0	0
41	224	0	0	0	0	0	0	0	0	0	0
42	1353	0	0	0	0	0	0	0	0	0	0
43	251	90	6	0	0	0	0	0	0	0	0
44	297	93	1	94	500	0	0	0	0	0	0
45	500	0	0	0	0	0	0	0	0	0	0
46	1100	100	1	0	0	0	0	0	0	0	0
47	52	0	0	0	0	0	0	0	0	0	0
48	190	0	0	0	0	0	0	0	0	0	0
49	166	0	0	0	0	0	0	0	0	0	0
50	375	0	0	0	0	0	0	0	0	0	0
51	10	20	1	0	735	142	0	0	0	0	0
52	64	0	0	0	0	0	0	0	0	0	0
53	1264	0	0	0	0	0	0	0	0	0	0
54	110	0	0	0	0	0	0	0	0	0	0
55	500	0	0	0	0	0	0	0	0	0	0
56	1064	0	0	0	0	0	0	0	0	0	0
57	64	20	1	0	420	0	0	0	0	0	0
58	116	9	500	0	0	0	0	0	0	0	0
59	110	9	530	0	0	0	0	0	0	0	0
60	105	57	54	0	0	0	0	0	0	0	0
61	120	0	0	0	0	0	0	0	0	0	0
62	385	20	1	0	1470	0	0	0	0	0	0
63	105	53	33	54	15	0	0	0	0	0	0
64	105	59	36	0	0	0	0	0	0	0	0

Figure II-3-C-3. Secondary Equipment for Blue

is a reflection of the data placed in data file 6. Across the top at far left is the title, PRIMARY EOH, and to the right is EOH, item code number, followed in the next column with AMT and so on. The first line states that the primary equipment item code number 21 has a secondary item in the quantity of 21. There are no other secondary items associated with item code 21. Formats of secondary equipment can be reviewed for validation. Should an error be found it may be traced directly to the 80-column card printout and corrected.

4. UNIT TYPE DESIGNATOR BREAKDOWN TABLE. The third type printout is that of unit type designator breakdown table for either Red or Blue forces illustrated in Figure II-3-C-4. This table is for the senior and subordinate unit relationship and reflects the data loaded onto data file 53. At the far left is the UTD for the senior unit. At the right of that column are the UTDs of the subordinate units assigned to that senior unit. Beneath each subordinate UTD is the number of these subordinate unit types that are assigned to the senior unit. Again, these may be used to check the accuracy of information entered in the data base. Apparent errors can be checked against the punch card 80-column printout and the errors on a specific card are detected.

5. UNIT TYPE DESIGNATOR (UTD) DIRECTORY. The fourth type printout for TOE constant data input is illustrated in Figure II-3-C-5. The purpose of this printout is to display the number of basic and complex units that are in the total force structure. This printout reflects the data entered onto data file 51. At the far left of the figure is the label index. To the right and along the top line is the number 142; this number has been assigned by the computer to the unit AALL. On the line labeled amount beneath AALL is the number 1; thus, in the entire force there is one AALL unit, and it has been assigned the index number of 142. Within this printout, the UTDs of all basic units are listed first--alphabetically sorted--followed by the complex units alphabetically sorted. The index numbers of the basic units provide a rapid reference to the printout of the TOE table.

6. AUTHORIZED PERSONNEL AND EQUIPMENT. The fifth type TOE data printout is illustrated in Figure II-3-C-6. For each basic UTD, the UTD index number and number of authorized personnel are shown at the left. Authorized equipment is shown in terms of the equipment item code, amount authorized to the unit, amount authorized on unit trains, and means of distribution (1 = unit, 2 = supply point distribution). The amount authorized to the unit and on trains is reflected on data file 50.

[illegible]

**Figure II-3-C-4. Unit Type Designator Breakdown Table**

.....

INDEX	05	61	02	63	32	26	24	25	30	117	110	95	112	31	94	133	86	83	84
PRASE	PCSF	PESE	PFPA	PICI	PIMI	PNCI	PNFA	PMNT	PNFA	POLL	PSLL	PSMH	PILL	PTST	PMFM	QBDE	QBSE	QDSE	QDSF
YVF																			
AMOUNT	2	6	2	1	3	9	6	2	10	1	1	3	1	3	2	1	2	1	6

**Figure II-3-C-5. Unit Type Designator Directory**

[illegible]

**Figure II-3-C-6. Authorized Personnel and Equipment Printout**

NOTES

II-3-C-8



APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR ORGANIZATION AND EQUIPMENT

(AVAILABLE UNDER SEPARATE COVER)

II-3-D-1

NOTES

II-3-D-2

## CHAPTER 4

### TASK ORGANIZATION DATA INPUT PROGRAMS

1. INTRODUCTION. The Task Organization data input programs create and load the unit status and supply status files and the authorized equipment file. The contents of these files are controlled by the Organization and Equipment data and the Task Organization data describing the unit resolution and command and support relationships among units.
2. ROUTINES. The Task Organization data input programs consist of the routines described below.
  - a. Routine ECHLON. The principal controlling routine is ECHLON. This routine creates and loads the unit status and authorized equipment files. It controls the reading and editing of appropriate data cards.
  - b. Routine RECOG1. The task organization data cards are read by routine RECOG1 and the data are returned in arrays stored in common.
  - c. Routine COMMCK. Routine COMMCK determines if a card read is a comment card. If not, it prints the card image. If a card is not terminated by a period, the next card is read.
  - d. Routine COMPRS. Blank spaces are compressed out of data card images by routine COMPRS.
  - e. Routine CFIND. Routine CFIND is a utility routine which searches a card image for a specified character and returns the location of a match if one is found.
  - f. Routine BLNKOT. A complete card image is filled with blanks by routine BLNKOT.
  - g. Routine CORDIN. Routine CORDIN searches a card image for a pair of X-Y coordinates and returns the coordinates in memory.
  - h. Routine SUPORT. Routine SUPORT reads support data and establishes the necessary support linkages. The support may be general support, direct support, reinforcing, or general support/reinforcing.
  - i. Routine SEQUEN. The routine that controls the creation of records for all units on the unit status file is SEQUEN. It calls routines TOEPUT, UPDATE, RELATR, and SUPPLY upon completion.
  - j. Routine TOEPUT. Routine TOEPUT fills the unit status record and authorized equipment record for a unit and places them on the data files.

k. Routine UPDATE. Updating of quantities of equipment and personnel on hand and authorized is performed by routine UPDATE for all units with subordinates. The quantities include the total of all quantities sent in subordinate units.

l. Routine CRAT31. Supply status records are created by routine CRAT31 and placed on the supply status file.

m. Routine RELATR. Routine RELATR inserts properly coded indications of supporting roles into all unit status records of a force.

n. Routine SUPPLY. The supply source for each unit is placed in the supply status records by routine SUPPLY.

o. Routine TALLY. Routine TALLY verifies that all task organization requirements have been satisfied.

3. FILES. The following data files are created and loaded by the Task Organization data input programs:

- . Unit status file - data file 1
- . Supply status file - data file 31
- . Authorized strength file - data file 50

## APPENDIX A

### TASK ORGANIZATION DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for task organization are documented in Appendix A, Unit Representation Input Requirements, to Chapter 3 of the Period Processor (Section IV).

NOTES

II-4-A-2

1 5 APR 1975

APPENDIX B

TASK ORGANIZATION DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. The task organization input data are loaded by the group of routines described in this appendix that are controlled by the routine ECHLON. For the set of data describing the task organization of one force, processing is accomplished in three passes. First, the basic organization input cards are read, recognized, and placed in a common storage area by RECOG1 and supporting utility routines called from ECHLON. Second, ECHLON calls the routine SEQUEN that constructs unit status records (data file 1) and supply status records (data file 31) using the input data. Finally, routine SUPPLY reads the supply point data and sets appropriate entries on the supply status records. Variables are passed from RECOG1 to SEQUEN with a set of arrays in common. The three common areas for this set of loading routines are as follows:

a. Common ONE (Labeled Area):

<u>Variable</u>	<u>Base Address</u>	<u>Contents</u>
IFNT(56,3)	1	File name table.
UID(2,600)	169	Unit identifications in load order.
ICLASS(200)	1369	Equipment type classes.
IDUM(200)	1569	Not used.
NBLUE	1769	Total number of blue units.
NRED	1770	Total number of red units.

b. Common TWO (Labeled Area):

<u>Variable</u>	<u>Base Address</u>	<u>Contents</u>
EVTBLE(4000)	1	Records 1 and 2 of data file 51 unit type designator and number allowable.
UIDTAB(2,1000)		Unit identifications in IUID order.

c. Blank Common (Unlabeled Area):

<u>Variable</u>	<u>Base Address</u>	<u>Contents</u>
LIST	1	List flag: 1 = list, 0 = no list.
NU	2	Total number of units in table.

1 5 APR 1975

<u>Variable</u>	<u>Base Address</u>	<u>Contents</u>
LEV(600)	3	Level (echelon) of unit.
ITOE(600)	603	Unit type designator tables.
XLOC(600)	1203	X coordinates of units.
YLOC(600)	1803	Y coordinates of units.
USUP(2,600)	2403	Unit being supported.
NSUP(600)	3603	Types of support.
MAXLEV	4203	Highest level (echelon) of a unit.
KEND	4202	Total number of records on data file 31.
I23	4205	Data file 23 pointer.
IRATE	4206	Consumption rate of food supplies.

2. ROUTINE ECHLON:

a. Purpose. ECHLON is the controlling routine in the sequence of routines designed to build a task organization including unit status records, supply status records, and unit authorized strength records.

b. Input:

- (1) Task organization data.
- (2) TOE data.
- (3) Supply data.

c. Output. ECHLON builds data file 1 (unit status), data file 31 (supply status), data file 50 (authorized strength), and data file 23 (unit composition).

d. Logical Flow (Figure II-4-B-1):

- (1) Block 1. Call GETFLE to get the file name table from disk and place it in common.
- (2) Block 2. Call REMOVE to remove data files 12, 23, and 31 from the file name table (IFNT) and remove the corresponding data from the disk storage area.
- (3) Block 3. Call CREATE to create a new file in IFNT for data files 12, 23, and 31.

II-4-B-2



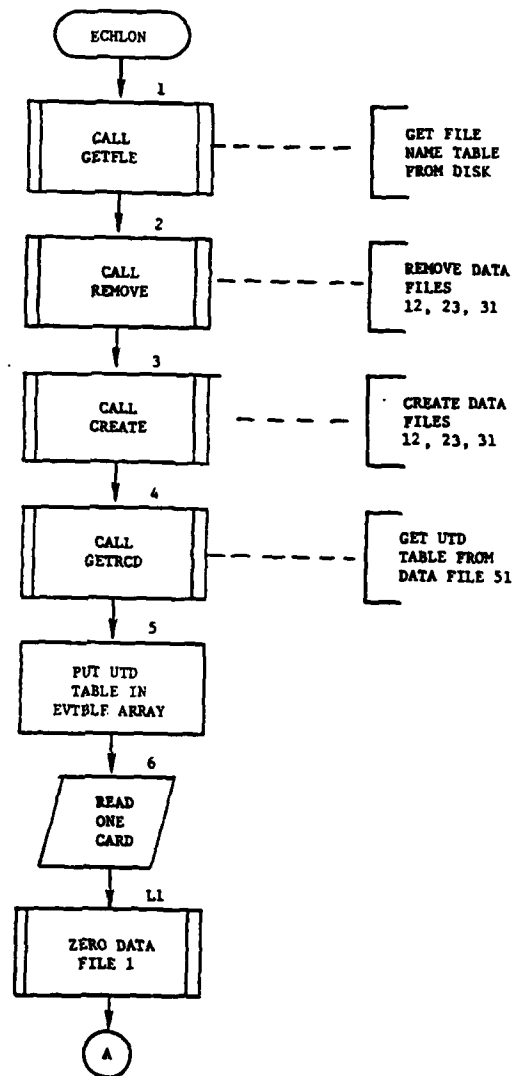


Figure II-4-B-1. Routine ECHLON (Continued on Next Page)

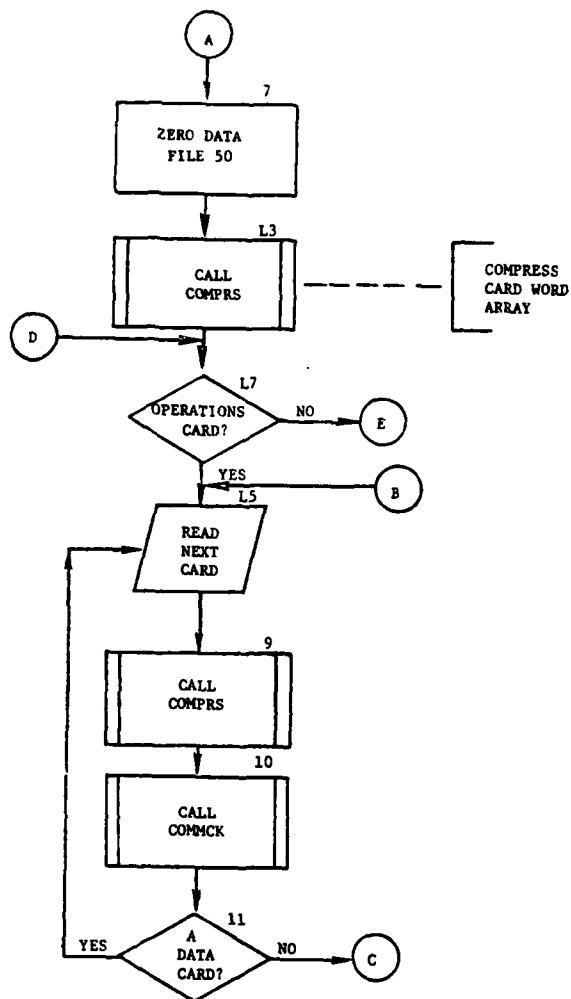


Figure II-4-B-1. Routine ECHLON (Continued)

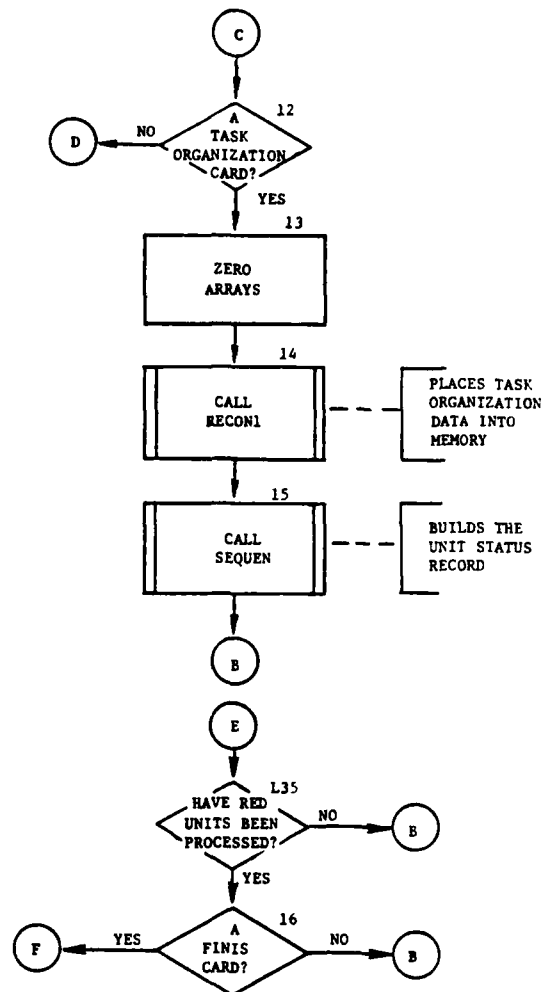


Figure II-4-B-1. Routine ECHLON (Continued)

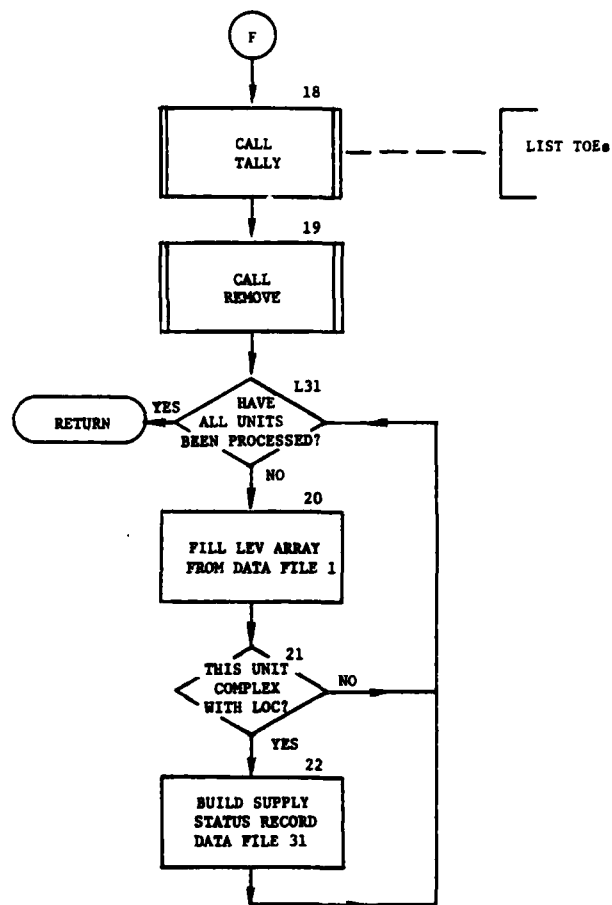


Figure II-4-B-1. Routine ECHLON (Concluded)

- (4) Block 4. Call GETRCD to retrieve the unit type designator table from data file 51.
- (5) Block 5. Put the unit type designator table into the EVTBLE array.
- (6) Block 6. Read one card from task organization.
- (7) Blocks L1 and 7. Zero the unit status file (data file 1) and the authorized strength file (data file 50).
- (8) Block 13. Call COMPRS to put card image into a compressed form.
- (9) Block L7. If this is not an operation card (first card), control goes to block L35.
- (10) Blocks L5 and 9. Read the next card and call COMPRS to compress the characters.
- (11) Blocks 10, 11, and 12. Call routine COMMCK to process this card if it is a comment card. If this is not a task card, control transfers to block L7.
- (12) Block 13. Blank arrays UID, XLOC, YLOC, USUP, and ITOE. Zero arrays LEV, NSUP, MAXLEV, and NU.
- (13) Block 14. Call RECOG1 to read the task organization data cards and place the input data into computer memory. If there were any errors in RECOG1 control transfers to block L5.
- (14) Block 15. Call SEQUEN to build the unit status file. When control returns from this routine, control transfers to block L5.
- (15) Block L35. Blue was processed with the first pass. If a second pass has not been made to process the Red units, return to block L5.
- (16) Block 16. If the last card read was not a FINIS card, control goes to block L5.
- (17) Block 18. Call TALLY which lists the units' TOEs.
- (18) Block 19. REMOVE is called to remove data file 12 from IFNT. Data file 12 passes the data from routine TOEPUT to UPDATE.
- (19) Block L31. If all units have been processed, control returns to the calling routine.
- (20) Block 20. Fill the LEV (echelon level of a unit) array from data file 1.
- (21) Block 21. If this unit does not have a location, control goes to block L31.

(22) Block 22. Build the supply status record for data file 31. Control goes to block L31.

3. ROUTINE RECOG1:

a. Purpose. RECOG1 reads task organization input data cards and places the input data into computer memory. Data are placed in a series of data arrays with the location within each array reflecting the sequence that data cards were read.

b. Input:

- (1) Task organization data cards.
- (2) Common blocks from routine ECHLON.

c. Output:

- (1) RECOG1 lists all data cards as they are read.
- (2) RECOG1 sets the following common variables: UID, ITOE, XLOC, YLOC, NSUP, UPSUP, LEV, MAXLEV, and NU.

d. Logical Flow (Figure II-4-B-2):

(1) Block L120, 1, and 2. A data card is read and routine COMPRS is called to compress any blanks out of the card image. Routine COMMCK is then called to determine if this is a comment card and to process it.

(2) Block 3 and L451. A check is made to determine if this is a support card. If it is, an error message is printed, the error flag is set, and control returns to the calling routine.

(3) Block 4. If there is not a period on the card, this unit's description is not complete, and control is transferred to block L120.

(4) Block 5. If this card is an end-of-unit card, control transfers to block L240.

(5) Block 6, 7, and 8. Set level indicator array for unit. If the new level is greater than the maximum level variable, reset the maximum level variable.

(6) Block 9. If this is a superior unit, transfer control to block L150.

(7) Block 10. If the level indicator of this basic unit equals one, transfer control to block 11.

(8) Block L150 and 13. If this is a superior unit, increment the level indicator.

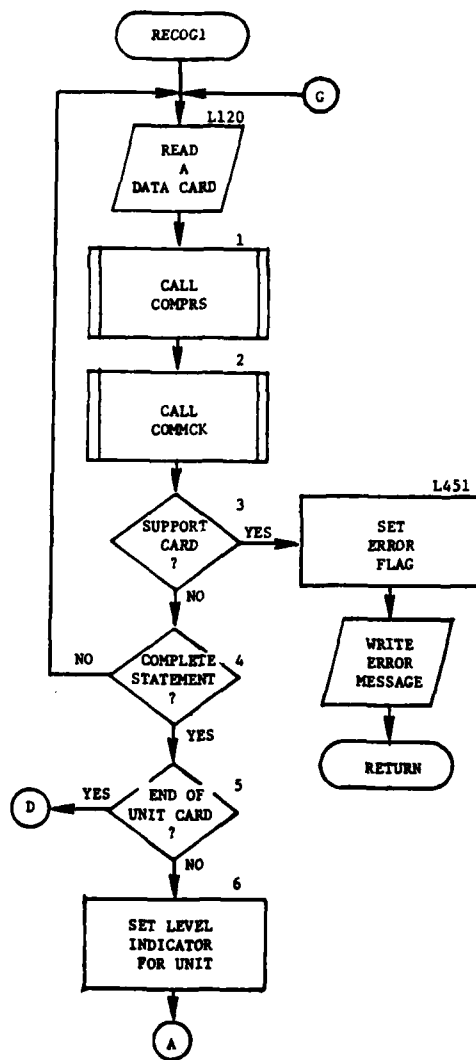


Figure II-4-B-2. Routine RECOG1 (Continued on Next Page)

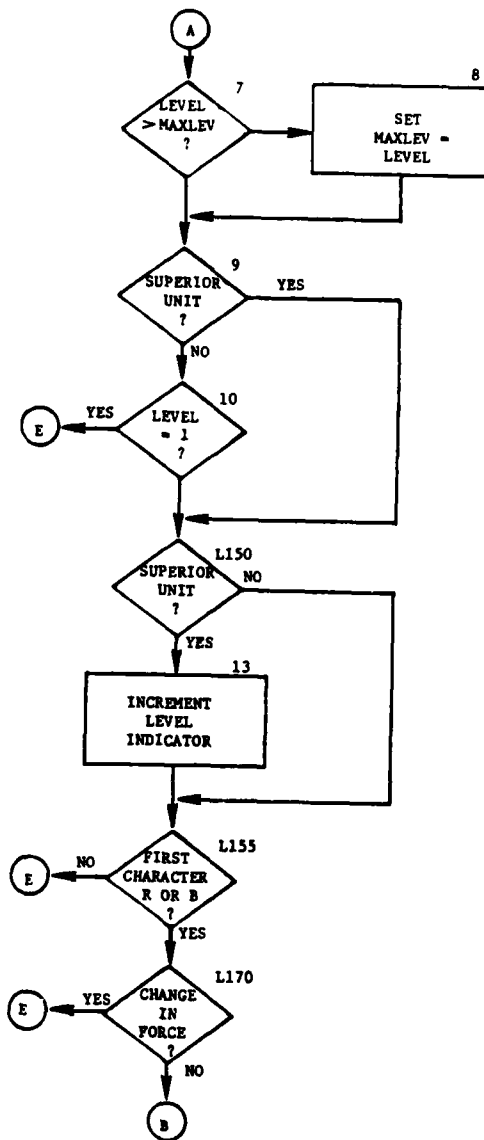


Figure II-4-B-2. Routine RECOG1 (Continued)



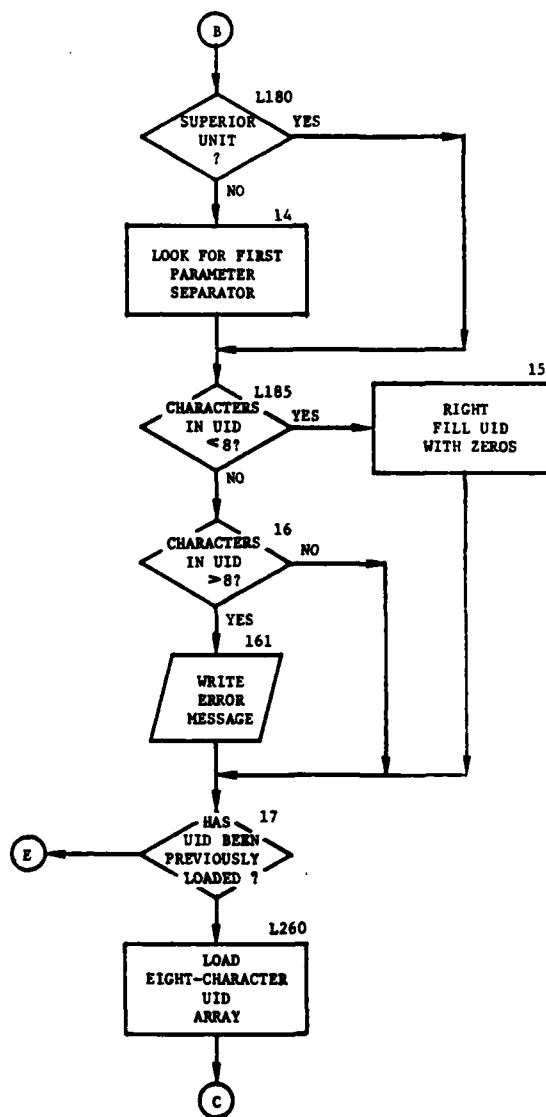


Figure II-4-B-2. Routine RECOG1 (Continued)

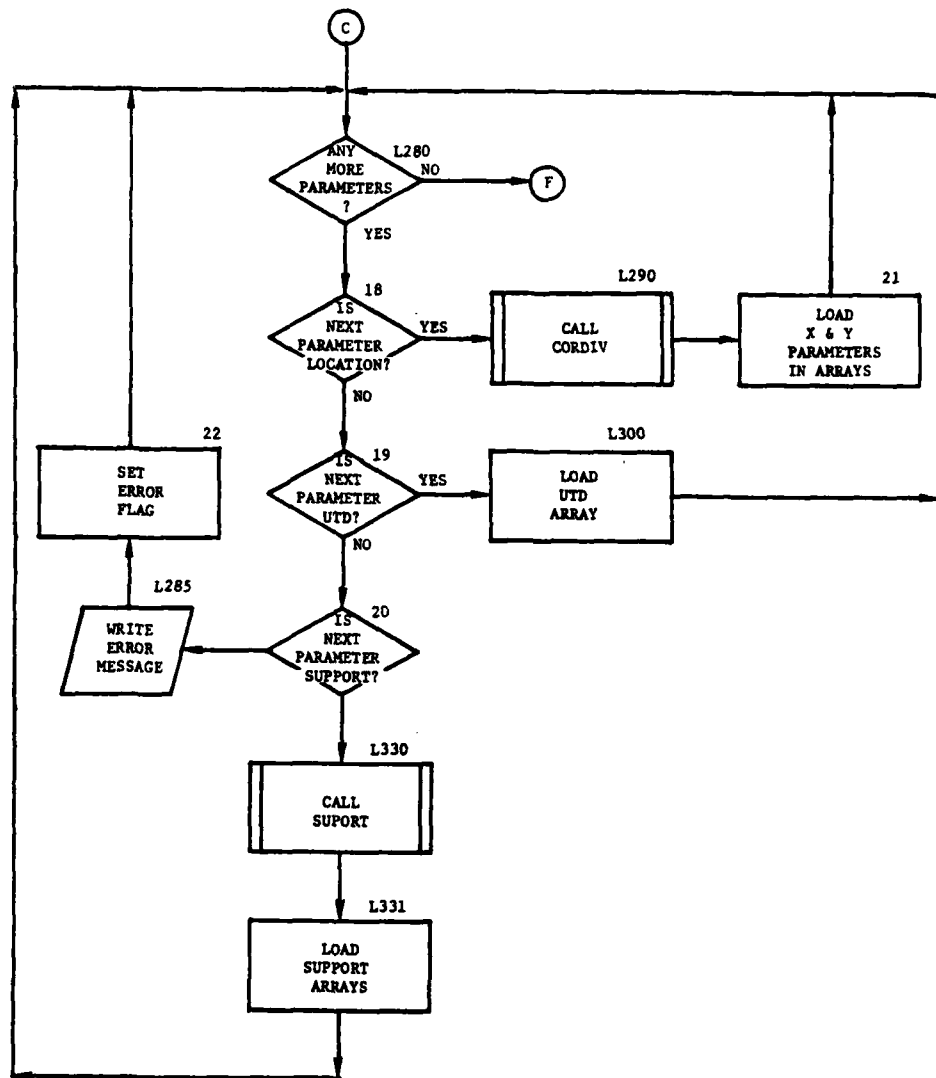
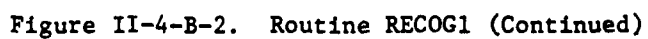


Figure II-4-B-2. Routine RECOG1 (Continued)



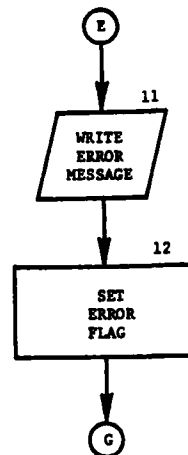
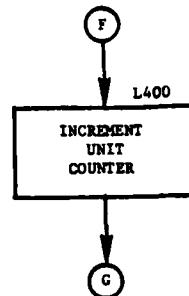


Figure II-4-B-2. Routine RECOG1 (Concluded)

(9) Block L155 and L70. If the first character of the unit identification (UID) is not B or R, or there has been a change in the force, transfer control to block 11.

(10) Block L180 and 14. If this is not a superior unit, locate first parameter separator in card.

(11) Block L185 and 15. If the number of characters in the UID is less than eight, write an informative message, right fill the UID with zeros, and transfer control to block 17.

(12) Block 16 and 161. If the number of characters in the UID is greater than eight, write an error message.

(13) Block 17. A check is made to determine if this UID has previously been loaded. If it has, transfer control to block 11.

(14) Block L260. Load the eight characters of UID into UID array.

(15) Block L280. A check is made to determine if there are any more parameters to process in the card image. If there are not, transfer control to block L400.

(16) Block 18, L290, and 21. If the next parameter is a location parameter, call routine CORDIN to extract the X and Y coordinates, load the coordinate arrays, and transfer control to block L280.

(17) Blocks 19 and L300. If the next parameter is a unit type designator parameter, load the parameter into the unit type designator array, and transfer control to block L280.

(18) Blocks 20, L285, and 22. If the next parameter is not a support parameter, write an error message, set the error flag, and transfer control to block L280.

(19) Blocks L330 and L331. Call routine SUPORT to extract the support data, load the data into the support arrays, and transfer control to block L280.

(20) Block L400. Increment the unit counter and transfer control to block L120.

(21) Block L420, 23, and 24. If the UID identified in the end-of-unit card has not been loaded write an error message, set the error flag and transfer control to block L120.

(22) Block L450. Decrement the level counter.

(23) Blocks 25 and 26. If the decremented level is not equal to the level of the unit in question, write an error message.

(24) Blocks 27 and L452. If the decremented level is not equal to one, transfer control to block L120; otherwise, list the UIDs, each associated

unit type designator, X and Y coordinates, support type, and level. Return control to the calling routine.

(25) Blocks 11 and 12. Write an error message, set the error flag, and transfer control to block L120.

#### 4. ROUTINE COMMCK:

a. Purpose. COMMCK reads from the character word array and identifies comment cards.

##### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
CCARD	Call	Compressed character array.
ICARD	Call	Word array.
JCHAR	Call	Last character column to search.
IGO	Call	Comment card flag.

c. Output. COMMCK compresses the card image and returns a flag if it is a comment card. If IGO equals 10, it is a comment card; if IGO equals 5, it is a data card.

##### d. Logical Flow (Figure II-4-B-3):

(1) Block 1. If the card is a comment card, set IGO equal to 10 and return control to the calling routine.

(2) Block L1. Print the card array on the printer in a 20A4 format.

(3) Block 2. If the last character of the card is a period, it is the end of data. Return control to the calling routine.

(4) Block 3. The end of data was not read; so, read another card image.

(5) Block 4. Call COMPRS to compress the next 80-column card image. Control goes to block L1.

#### 5. ROUTINE COMPRS:

a. Purpose. COMPRS puts cards into compressed form for up to 80 compressed characters.

##### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ICARD	Call	Card image array (four characters per word).
CCARD	Call	Character array.
JCHAR	Call	Number of compressed characters.

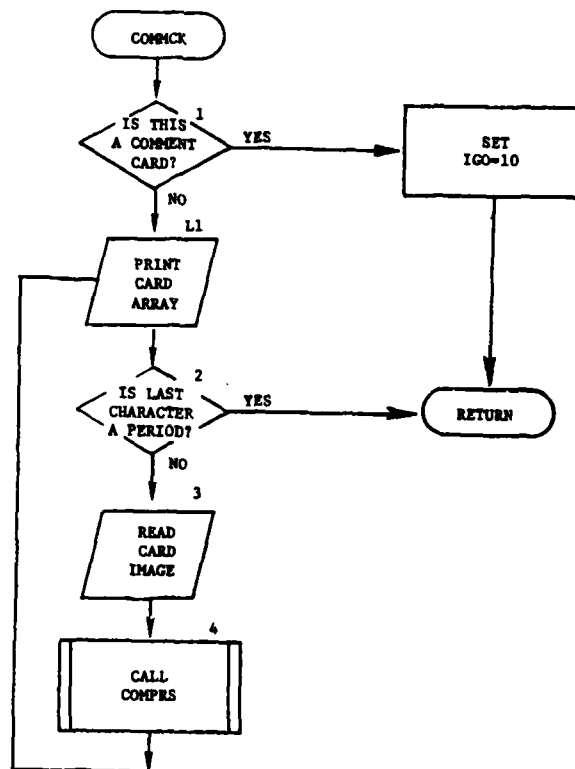


Figure II-4-B-3. Routine COMMCK

c. Output. Puts the nonblank characters into an array.

d. Logical Flow (Figure II-4-B-4):

(1) Block 1. The loop that passes through each word of the card is initialized or incremented.

(2) Block 2. The loop that passes through each character of the card word is initialized or incremented.

(3) Block 3. Call routine MVCHAR to extract the selected character of the word and place it left justified in a test variable.

(4) Block 4. If the test variable is blank, transfer control to block 7.

(5) Blocks 5 and L20. If the total number of characters in this statement exceeds 80, write an error message and return control to the calling routine.

(6) Block 6. Increment a character counter and move the character into the character array.

(7) Block 7. If this is not the last character of the word, transfer control to block 2.

(8) Block 8. If this is not the last word of the card, transfer control to block 1; otherwise, return control to the calling routine.

#### 6. ROUTINE CFIND:

a. Purpose. CFIND searches for a character and returns the column number it is in. If the character is not found, it returns 100.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
CCARD	Call	Compressed character array to search.
KK	Call	Starting column for search.
LCHAR	Call	Last character column to search.
KCHAR	Call	Character to search for.

c. Output. CFIND sets LCHAR to the number of the column where the character is found. If the character is not found, set LCHAR to 100.

d. Logical Flow:

(1) Search the character array for the character requested.

(2) If the character is found, let LCHAR equal column number and return control to the calling routine.

(3) If the character is not found, let LCHAR equal 100, and return control to the calling routine.



AD-A112 658

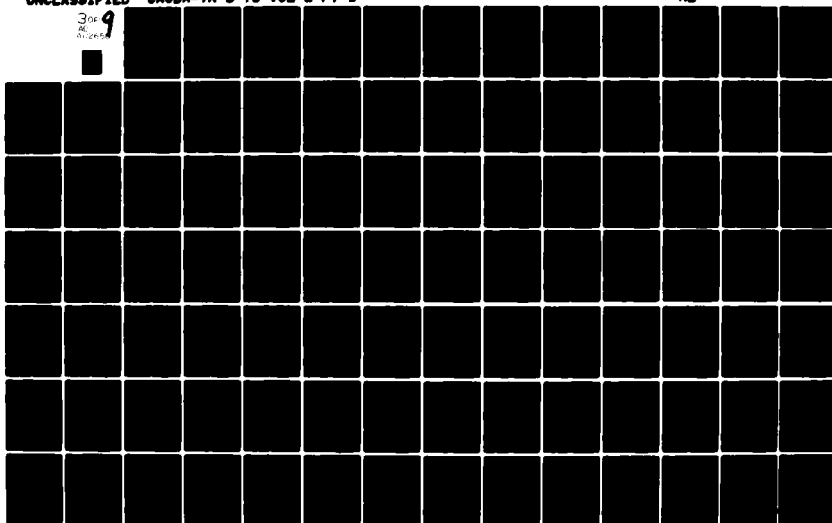
ARMY COMBINED ARMS-COMBAT DEVELOPMENTS ACTIVITY FORT--ETC F/O 18/7  
DIWAS MODEL DOCUMENTATION, VOLUME II. PROGRAMMER/ANALYST MANUAL--ETC(U)  
JUL 76

UNCLASSIFIED

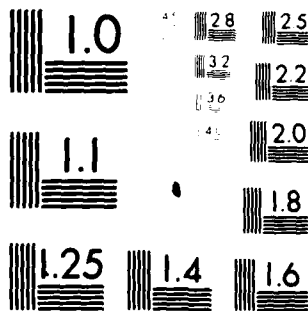
CACDA-TR-8-76-VOL-2-PT-1

ML

3 of 9  
ML  
ALC-112



11265



MICROCOPY RESOLUTION TEST CHART  
NATIONAL BUREAU OF STANDARDS-1963-A

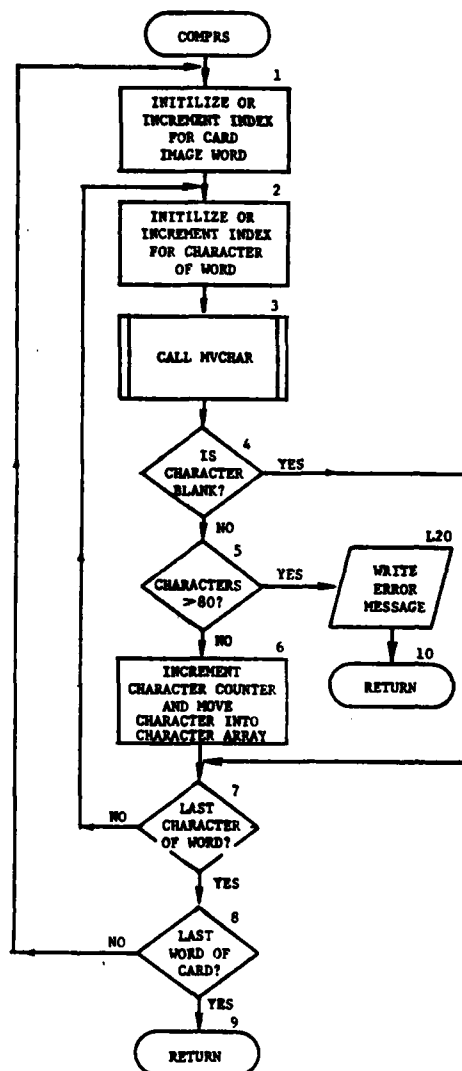


Figure II-4-B-4. Routine COMPRS.

7. ROUTINE BLNKOT:

a. Purpose. BLNKOT places blanks in a card image array.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ICARD	Call	The card array to be blanked.

c. Output. Returns a blank 20-word array to the routine.

d. Logical Flow:

- (1) Set ICARD(20) equal to blanks.
- (2) Return control to calling routine.

8. ROUTINE CORDIN:

a. Purpose. CORDIN places the card image of X and Y coordinates into memory storage locations.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
CCARD	Call	Character array.
JCHAR	Call	Number of characters.
K	Call	Starting column for search.
X	Call	X coordinate.
Y	Call	Y coordinate.

INFT(56,3), NBLUE, and NRED, and BLANK from common ONE are also input.

c. Output. CORDIN places the X and Y coordinates from the CCARD array onto memory.

d. Logical Flow (Figure II-4-B-5):

(1) Blocks 1 and 2. Find the beginning of the X coordinate in the character array. Call CFIND to find the end of the X coordinate. This is done by checking for the dash (-) character.

(2) Block 3. Place the X coordinate in memory.

(3) Blocks 4 and 5. Call CFIND to search for the end of the Y coordinate. When an asterick (\*) is found, the Y coordinate is placed onto memory and control returns to the calling routine.

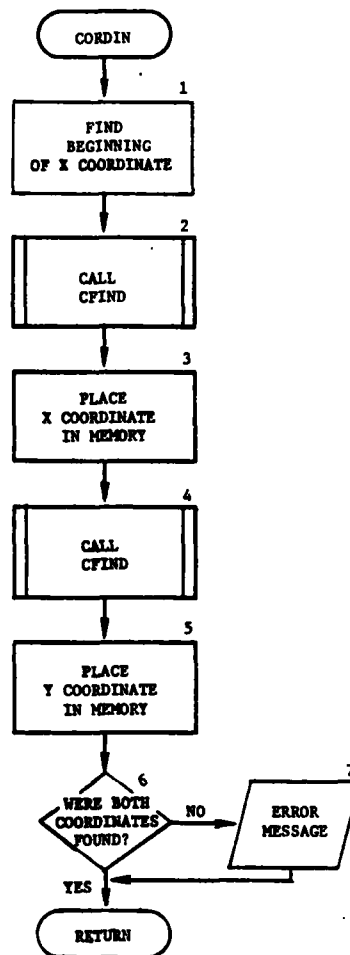


Figure II-4-B-5. Routine CORDIN

(4) Blocks 6 and 7. If no coordinates are found an error message is printed and control returns to the calling routine.

9. ROUTINE SUPORT:

a. Purpose. SUPORT decodes data to determine the unit receiving support and the type of support it is to receive.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
CCARD	Call	Compressed character array.
K	Call	Starting character column in compressed array.
JCHAR	Call	Number of compressed characters.

c. Output. SUPORT returns the type of support and the unit to receive it through the variable UNIT. The type of support is returned through K as follows:

- . If K = 0, general support
- . If K = 1, direct support
- . If K = 2, reinforcing
- . If K = 3, general support/reinforcing.

d. Logical Flow (Figure II-4-B-6):

(1) Block 1. If the support type is direct, transfer control to block L50.

(2) Block 2. If the support type is reinforcing, transfer control to block L70.

(3) Block 3. If the support type is general, transfer control to block L25.

(4) Blocks 4 and 5. If the support type is not general support/reinforcing or the supported unit is of the opposing force, transfer control to block 11.

(5) Blocks 6 and 7. Load the character representation of the supported unit into a temporary array, set the support type flag equal to three, and transfer control to block L90.

(6) Blocks L50 and 8. If there is a syntax error in the support unit or the supported unit is of the opposing force, transfer control to block 11.

(7) Blocks 9 and 10. Load the character representation of the supported unit into a temporary array, set the support type flag equal to one, and transfer control to block L90.

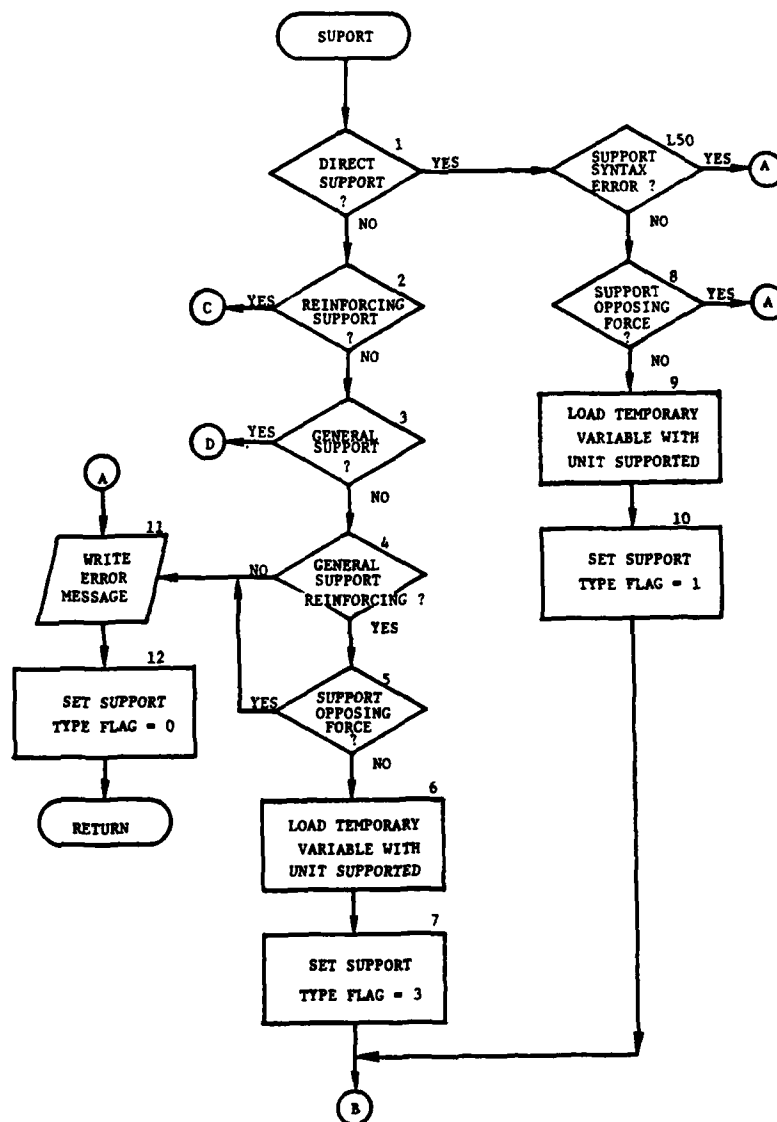


Figure II-4-B-6. Routine SUPPORT (Continued on Next Page)

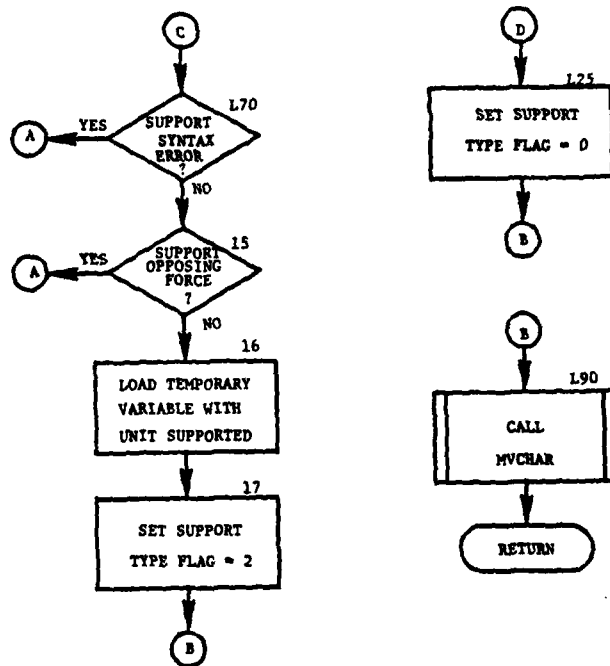


Figure II-4-B-6. Routine SUPORT (Concluded)



(8) Blocks 11 and 12. Write an error message, set the support type flag equal to zero, and return control to the calling routine.

(9) Blocks L70 and 15. If there is a support type syntax error or if the supported unit is of the opposing force, transfer control to block 11.

(10) Blocks 16 and 17. Load the character representation of the supported unit into a temporary array, set the support type flag equal to two, and transfer control to block L90.

(11) Block L25. Set the support type flag equal to zero.

(12) Block L90. Call routine MVCHAR to transfer the supported unit from the temporary array to the UNIT variable array, and return control to the calling routine.

#### 10. ROUTINE SEQUEN:

a. Purpose. SEQUEN is the main routine that builds the unit status file.

b. Input Variables:

(1) Common ONE and blank common area from ECHLON.

(2) Supply class data from data file 11.

c. Output. SEQUEN places all units and subordinates on the unit status file.

c. Logical Flow (Figure II-4-B-7):

(1) Block 1. Set the number-of-units counter to zero. Initialize UID temporary storage variable with first UID of force.

(2) Block 2. Set the force flag to 428 for the Blue force or 429 for the Red force.

(3) Blocks 3 and L10. Call routine GETWRD twice: once, to retrieve the supply classes from data file 11; and once, to retrieve the consumption rates from data file 11.

(4) Block L105. Set the start point for the level search to one.

(5) Block 4. If the MAXLEV variable is equal to one, all units have been processed and control transfers to block L180.

(6) Blocks L100 and 5. A search is made from the start point through the total number of units for a unit of which the level is equal to the MAXLEV variable. If one is not found, transfer control to block L165.

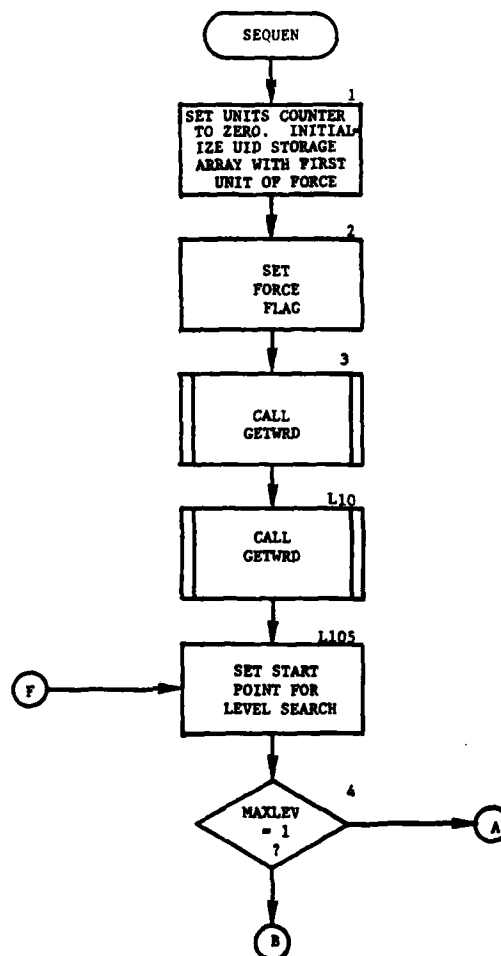


Figure II-4-B-7. Routine SEQUEN (Continued on Next Page)

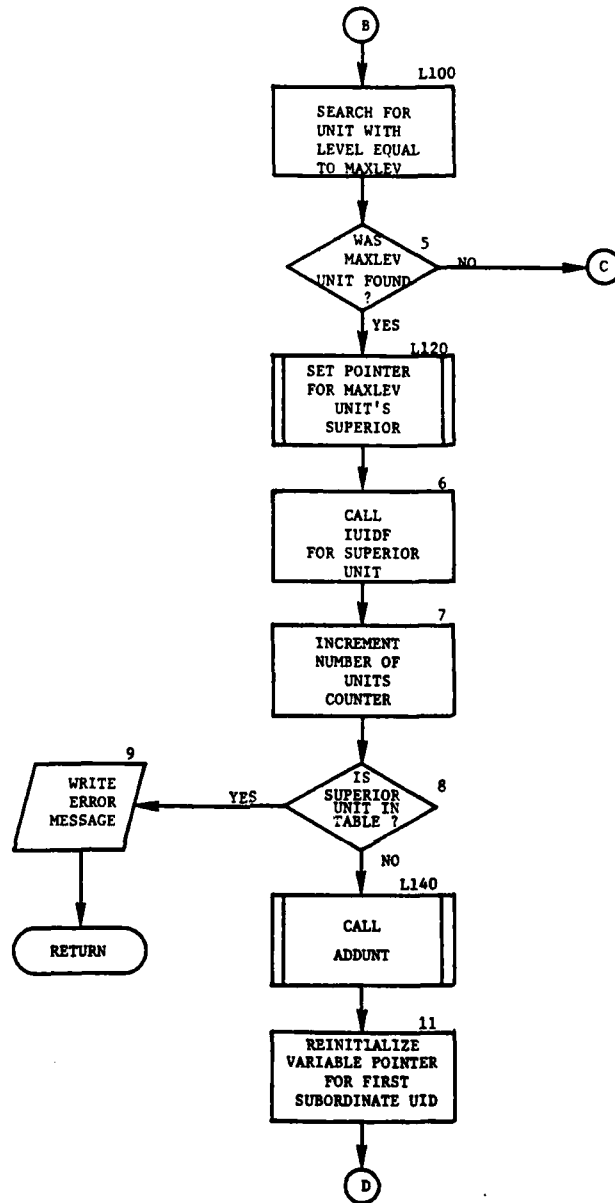


Figure II-4-B-7. Routine SEQUEN (Continued)

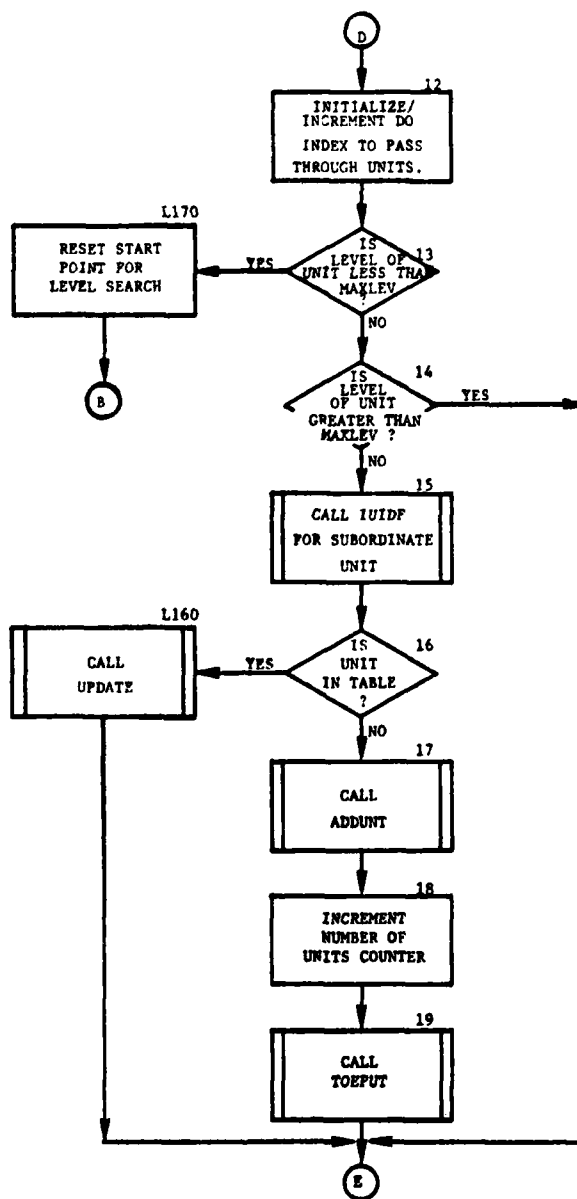


Figure II-4-B-7. Routine SEQUEN (Continued)

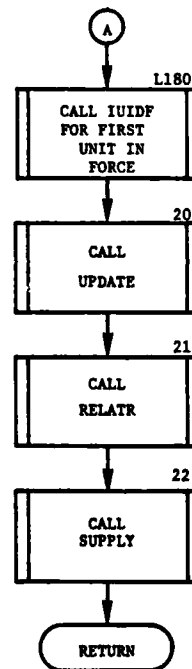
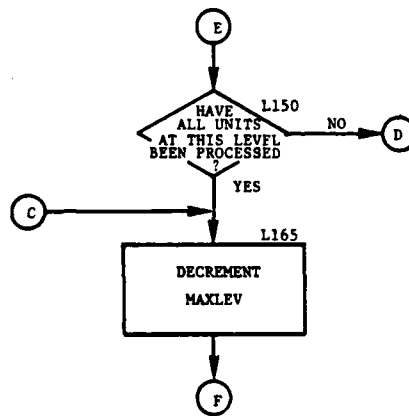


Figure II-4-B-7. Routine SEQUEN (Concluded)

(7) Blocks L120, 6, and 7. Set pointer for superior unit of maximum level unit found, call routine IUIDF to determine if the superior unit has previously been placed in the UID table, and increment the number-of-units counter.

(8) Blocks 8, 9, and 10. If the superior UID has previously been placed in the UID table, an error condition exists; there being two units with the same UID. An error message is written and control returns to the calling routine.

(9) Blocks L140 and 11. Routine ADDUNT is called to add the superior unit to the UID table, and the pointer for the subordinate UID with the maximum level is reinitialized.

(10) Block 12. Initialize or increment do index to pass through the force from the first maximum level unit to the end of the force.

(11) Blocks 13 and L170. If the level of the unit being examined is less than MAXLEV, reset the start point for the level search to that of the present unit, and transfer control to block L100.

(12) Block 14. If the level of the unit being examined is greater than the maximum level variable, transfer control to block L150.

(13) Blocks 15 and 16. Call routine IUIDF with the present subordinate UID as an argument to determine if it has previously been loaded in the UID tables. If it has, a call is made to routine UPDATE to update its superior unit's status record, and control is transferred to block L150.

(14) Blocks 17, 18, and 19. A call is made to routine ADDUNT to add the present subordinate unit UID to the UID tables, the units counter is incremented, and a call is made to TOEPUT to load unit status and authorized strength records for this unit.

(15) Blocks L150 and L165. If all the units of this level have been processed, decrement the MAXLEV variable and transfer control to block L105; otherwise, transfer control to block 12.

(16) Block L180. A call is made to routine IUIDF to retrieve the IUID of the first unit in the force.

(17) Blocks 20, 21, and 22. A call is made to routine UPDATE to load the first unit in the force's unit status record, a call is made to routine RELATR to update the supporting roles of the units in the unit status record; a call is made to routine SUPPLY to load the supply data into the unit status record; and, control returns to the calling routine.

11. ROUTINE ADDUNT:

a. Purpose. ADDUNT places the UID of a newly defined unit into a space available within the UID versus IUID cross reference table and returns the IUID allocated for the unit.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
UID	Call	Unit identification code.
IUID	Call	Integer unit identification.
BCOUNT	ONE	Number of Blue units in cross reference table.
RCOUNT	ONE	Number of Red units in cross reference table.

c. Output. ADDUNT returns the IUID allocated for the new unit to the calling routine.

d. Logical Flow. Set the appropriate IUID of the Red or Blue force unit to be allocated and place the UID in the cross reference table. Return control to the calling routine.

12. ROUTINE TOEPUT:

a. Purpose. TOEPUT fills the unit status and authorized equipment records for a basic unit.

b. Input Variables:

(1) Common ONE and blank common areas.

(2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IA	Call	Location in common of data for unit to be treated.
NREC	Call	Record number within data file 1 allocated for unit's status record.
IKP2	Call	Record number within data file 1 allocated for the unit's superior.
EVTBLE	TWO	Unit type designator table.

c. Output. TOEPUT outputs the unit status record on data file 1 and the authorized strength record on data file 50 for the unit created.

d. Logical Flow (Figure II-4-B-8):

(1) Block 1. The UID, UTD, and X and Y coordinates of the unit are loaded into their appropriate places in the data file 1 record array.

(2) Blocks 2 and 3. The UTD index table of data file 51 has been loaded into the variable array EVTBLE. The part of this table that holds basic UTDs is searched for a corresponding UTD. The ordinal of this match will give an index to data file 52 to retrieve equipment on-hand and in trains data. If a corresponding UTD is found, transfer control to block L30.

(3) Blocks L261 and 4. If processing goes to this block, the UTD of the unit is complex, and the complex part of the UTD table is searched for a corresponding UTD. The ordinal of this corresponding UTD gives an index to data file 53 to retrieve a breakdown of the complex UTD. If a corresponding UTD is not found, transfer control to block L263.

(4) Blocks L270, 5, and L271. If the original UTD of the unit is being processed, that UTD and the numeral 1 are loaded into a UTD and count array (TALLY array) that contains all UTD types and amounts used to build a unit.

(5) Blocks 6, 7, 8, and 9. When a UTD must be separated into basic UTDs a data file 23 record is built for that unit. If the data file 23 record has not previously been built for this unit, the data file 23 record array is loaded and a call is made to routine PUTLCD to place the record onto data file 23. The pointer to that record is placed into the data file 1 record array.

(6) Blocks L272 and 10. A do loop is established to access the entries within the retrieved data file 53 record. The odd-numbered words of a data file 53 record are the authorized UTD types, the even-numbered words are the amounts authorized. If the present UTD entry is blank, all UTDs in the record have been accessed and control transfers to block L280.

(7) Blocks 11 and 12. A pointer for a UTD holding array and the corresponding array that contains the different UTD types is incremented. The UTD is loaded into that position in the array. The number of such UTDs is multiplied by a multiplication factor, derived from the number of units superior to the authorized unit, and is loaded into the count array.

(8) Blocks 13, L277, and 14. If the present UTD has been previously processed for this unit, the TALLY array count is updated for that unit; otherwise, the UTD is loaded into the UTD TALLY array and the count is loaded into the count TALLY array.

(9) Block L280. If processing of the current data file 53 record is not completed, transfer control to block L272.



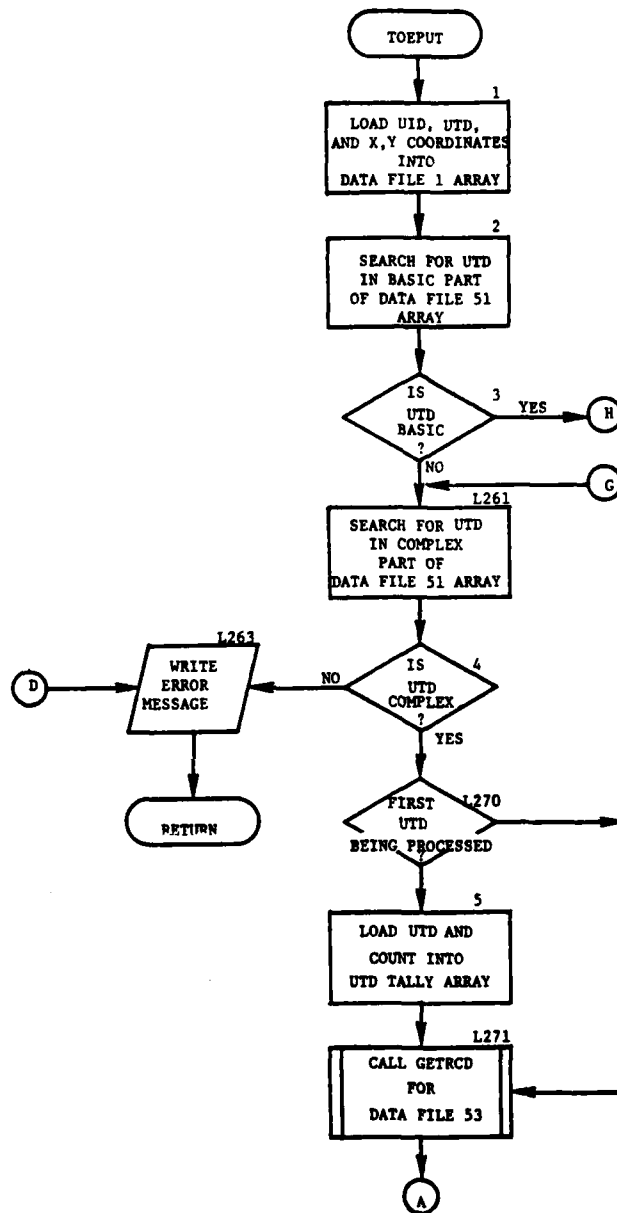


Figure II-4-B-8. Routine TOEPUT (Continued on Next Page)

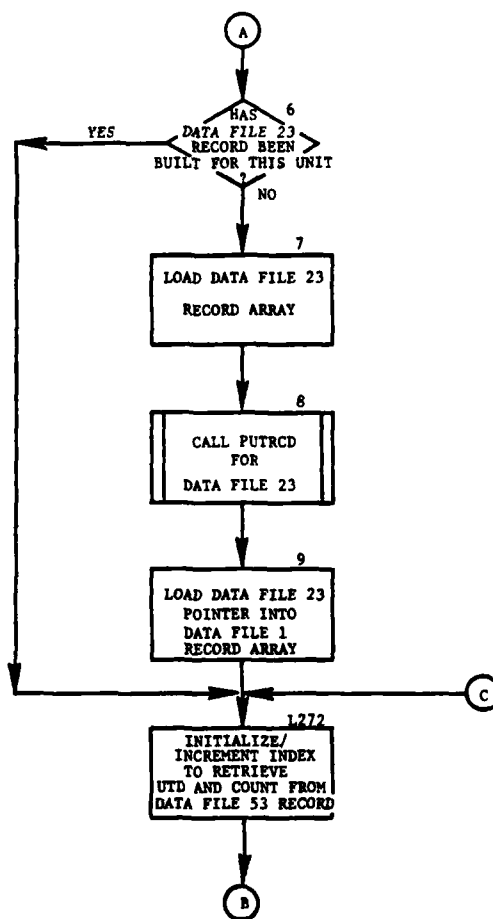


Figure II-4-B-8. Routine TOEPUT (Continued)

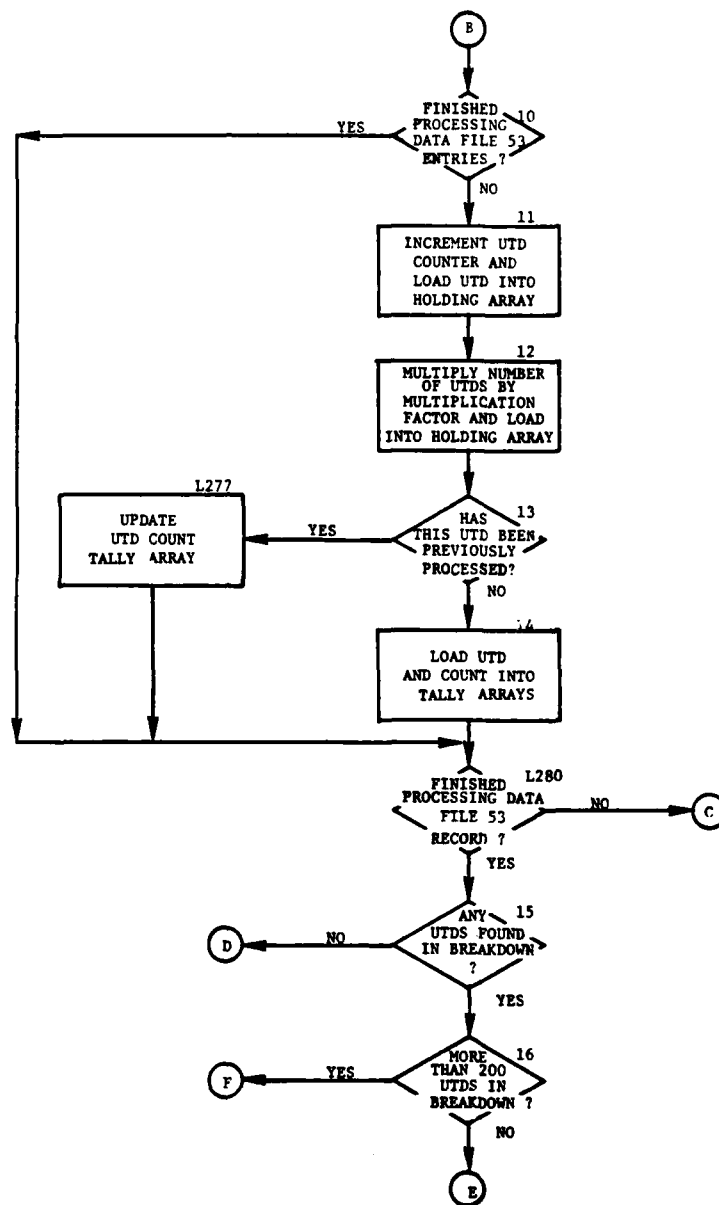


Figure II-4-B-8. Routine TOEPUT (Continued)

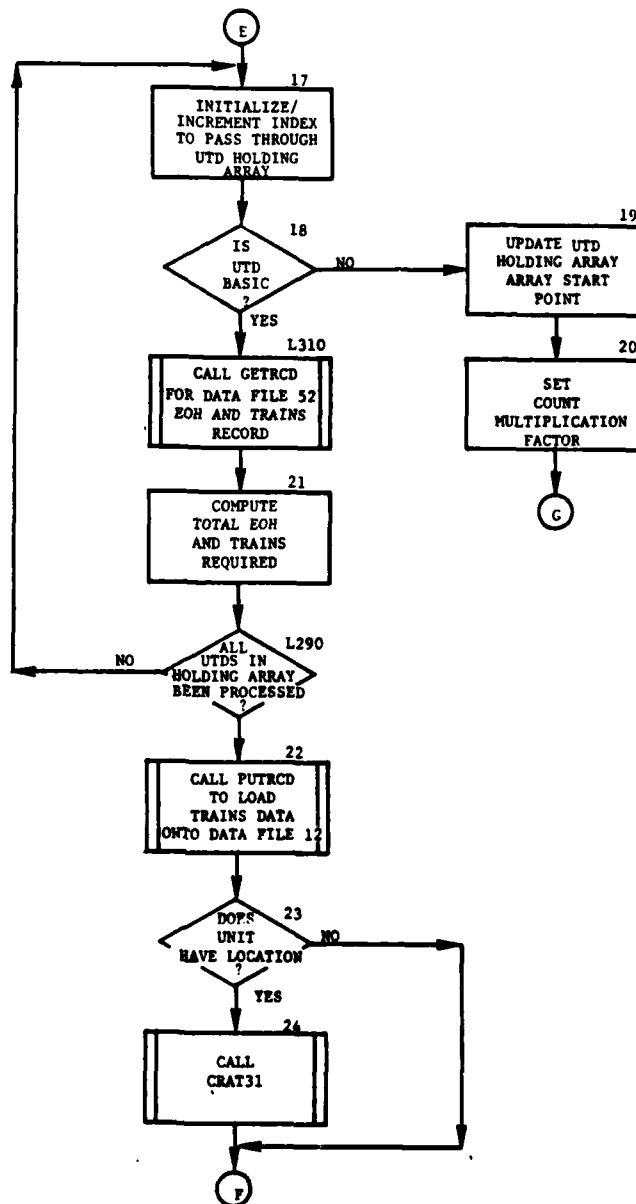


Figure II-4-B-8. Routine TOEPUT (Continued)

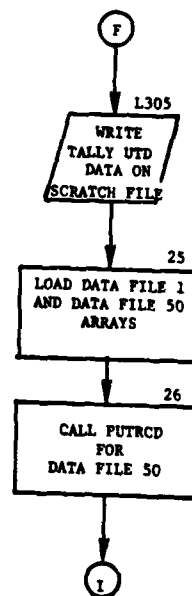


Figure II-4-B-8. Routine TOEPUT (Continued)

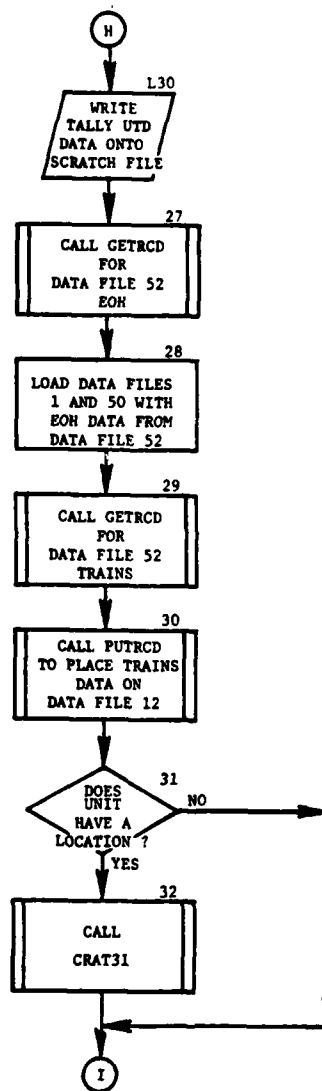


Figure II-4-B-8. Routine TOEPUT (Continued)

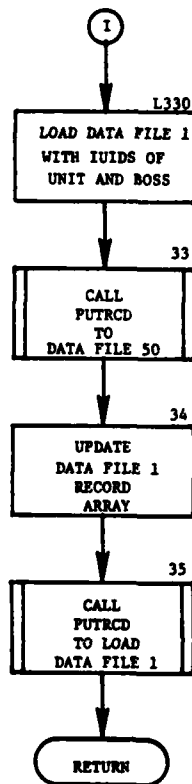


Figure II-4-B-8. Figure TOEPUT (Concluded)

(10) Blocks 15 and 16. If the complex UTD for this unit does not reduce to more basic units, transfer control to block L263. If the complex UID for this unit breakdown is to more than 200 basic units, transfer control to block L305.

(11) Block 17. At this point at least one record has been retrieved from data file 53 and placed in a holding array. This array indicates that the UTD is divided into more basic UTDs. A do index is initialized or incremented to pass through this array.

(12) Blocks 18, 19, and 20. If a UTD is found in the holding array that is not a basic UTD, the holding array start point is updated, the count multiplication factor is set, and control is transferred to block L261.

(13) Blocks L310 and 21. Call routine GETRCD to retrieve the data file 52 equipment-on-hand and in-trains data for this UTD and update total equipment-on-hand required and in trains.

(14) Block L290. If all UTDs in the holding array have not been processed, transfer control to block 17.

(15) Blocks 22, 23, and 24. A call is made to routine PUTRCD to load the trains data onto data file 12, and if the unit has a location, a call is made to routine CRAT31 to create data file 31 records for this unit.

(16) Blocks L305, 25, and 26. The TALLY data are written onto a scratch file. The appropriate data are loaded into the data file 1 and data file 50 record arrays, and a call is made to PUTRCD to output a data file 50 record. Control is then transferred to block L330.

(17) Blocks L30, 27, and 28. The TALLY unit type designator data are written onto the scratch file. A call is made to routine GETRCD to retrieve the appropriate data file 52 on-hand equipment record and that data are loaded into the data file 1 record array.

(18) Blocks 29 and 30. A call is made to routine GETRCD to retrieve the appropriate data file 52 trains data and then a call is made to routine PUTRCD to load that data into data file 12.

(19) Blocks 31 and 32. If the unit in question has a location, call routine CRAT31 to create the appropriate data file 31 records for that unit.

(20) Block L330. Load the data file 1 record array with the IUID of the unit and its superior unit.

(21) Blocks 33, 34, and 35. Call routine PUTRCD to load data file 50, update data file 1 record array, and call PUTRCD to load data file 1.

(22) Block L263. Write an error message and return control to the calling routine.



13. ROUTINE UPDATE:

a. Purpose. UPDATE fills the unit status record (data file 1) and the authorized equipment strength record (data file 50) of a unit that has subordinate units and adjusts unit status records of subordinate units if required.

b. Input Variables:

(1) Common ONE and blank common area.

(2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IA	Call	Location in common of data for unit to be treated.
NREC	Call	Record number within data file 1 allocated for unit's status record.
IKP2	Call	Record number within data file 1 allocated for the unit's superior.
IFORCE	Call	Indicates Red or Blue unit.
NI	Call	Number of subordinates pointer.
EVTBLE	TWO	Unit type designator (UTD) tables.

c. Logical Flow. (Figure II-4-B-9):

(1) Blocks 1 and 2. Initialize all variables and arrays to be used in this routine. Set the level parameters of the unit to be updated and the unit's immediate subordinates.

(2) Block 3. Set up a loop that will pass from the first unit after the unit to be updated to the end of the force.

(3) Block 4. If the next unit has a level less than or equal to the level of the updating unit (BOSS) it is not subordinate to that unit, and control transfers to block L37.

(4) Block 5. If the next unit has a level which is greater than that of its immediate subordinates this is a new echelon, then transfer control to block L30.

(5) Blocks 6 and 7. Call routine IUIDF to retrieve the IUID of the subordinate unit, and call routine GETRCD to retrieve the unit status record of that unit.

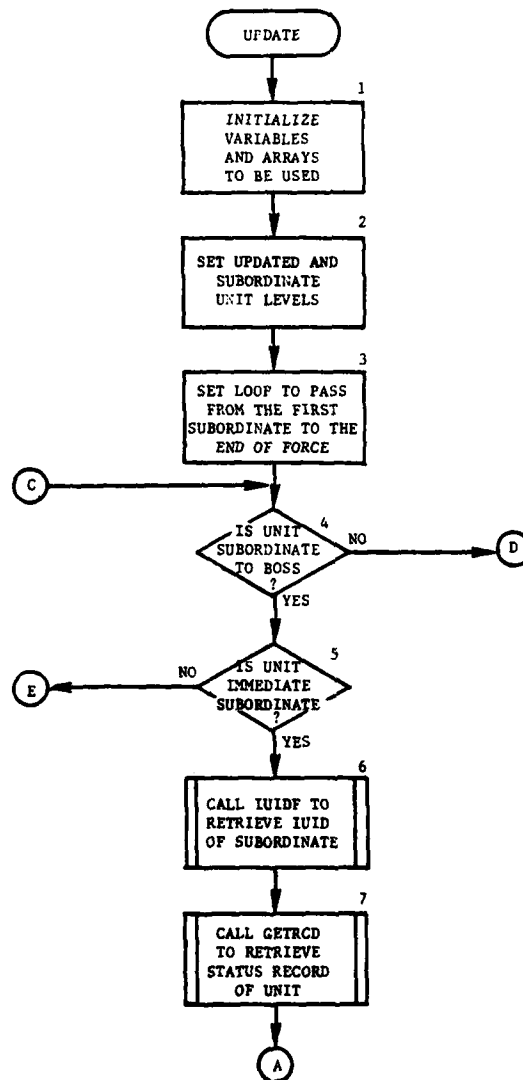


Figure II-4-B-9. Routine UPDATE (Continued on Next Page)

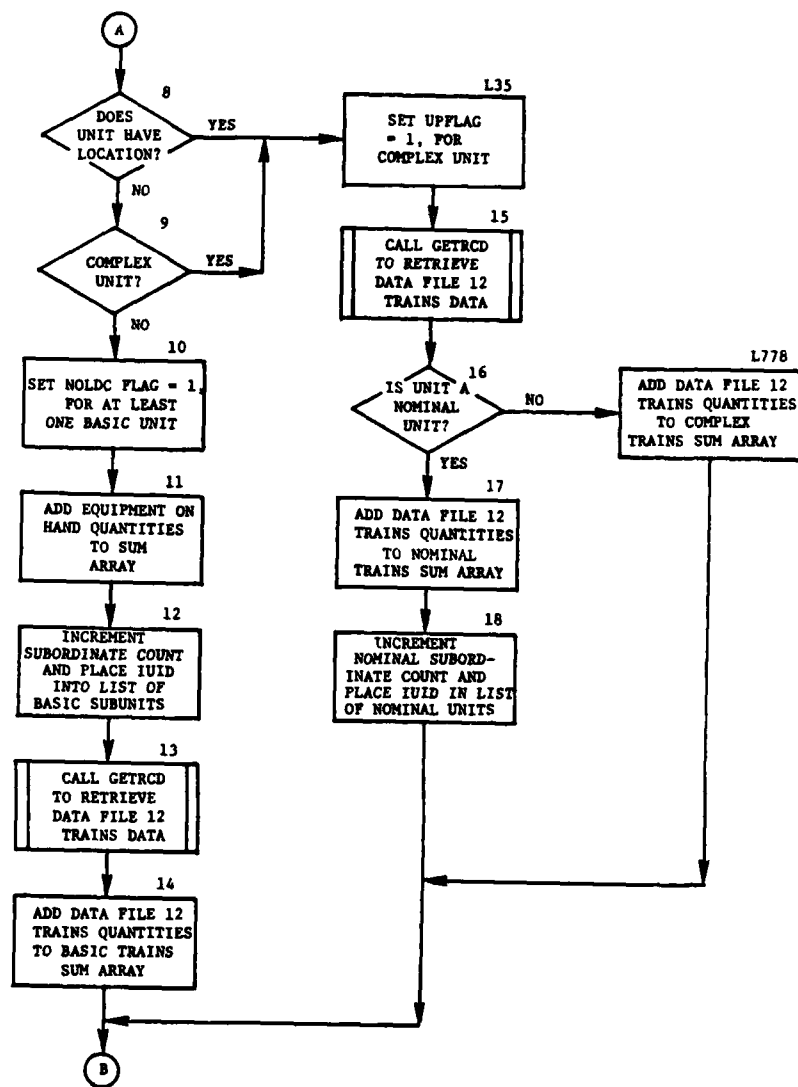


Figure II-4-B-9. Routine UPDATE (Continued)

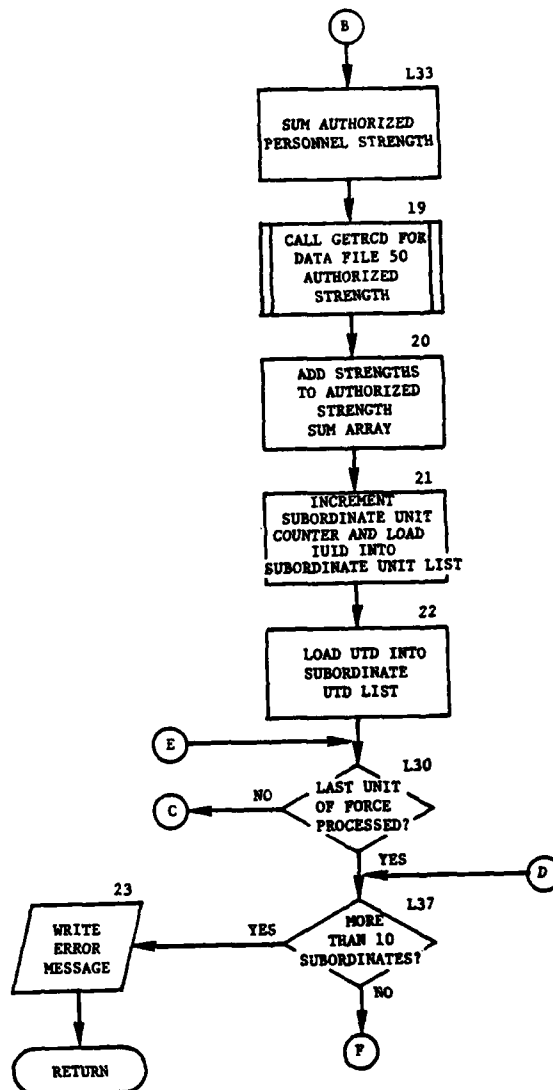


Figure II-4-B-9. Routine UPDATE (Continued)

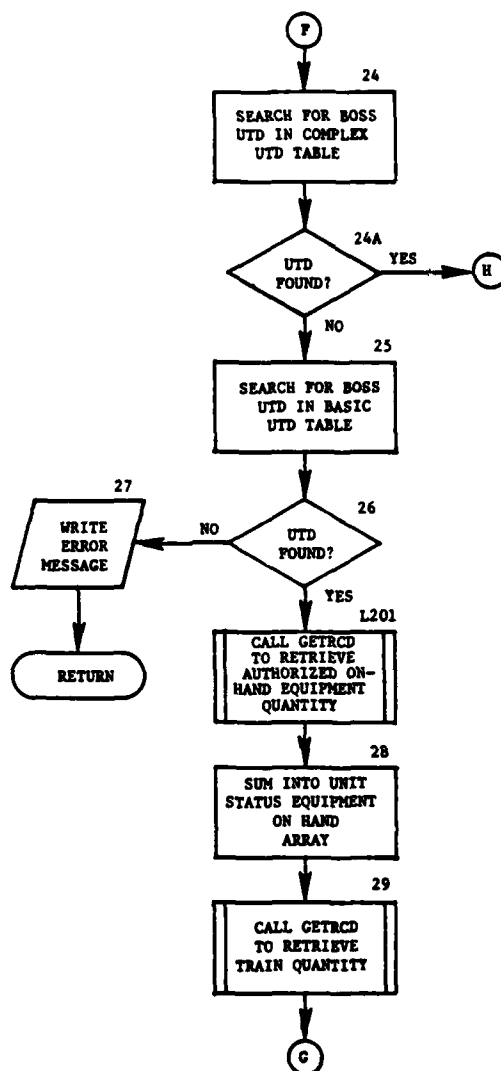


Figure II-4-B-9. Routine UPDATE (Continued)

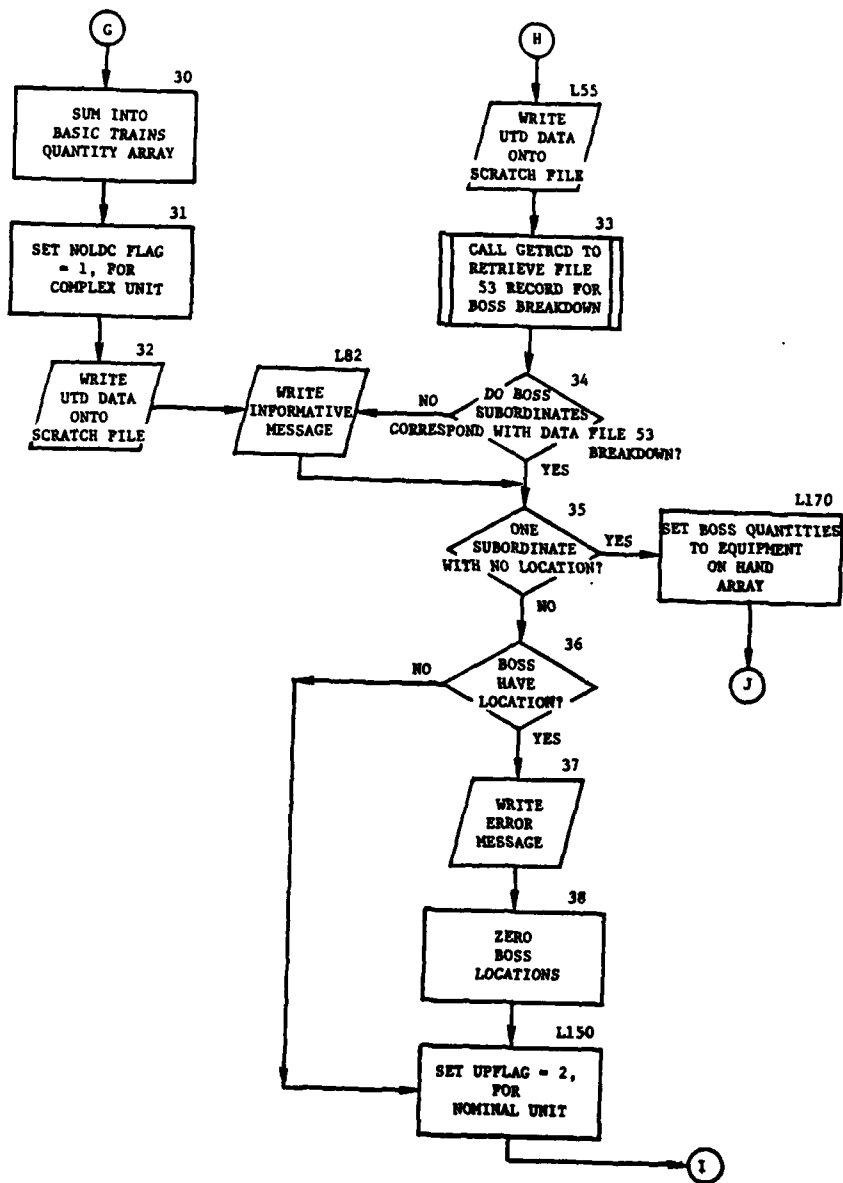


Figure II-4-B-9. Routine UPDATE (Continued)

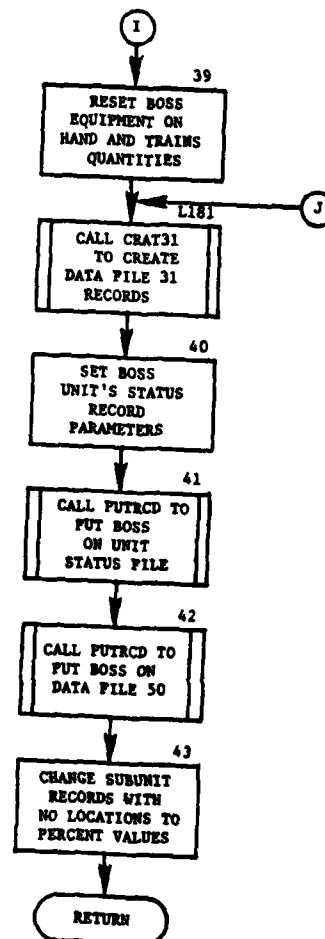


Figure II-4-B-9. Routine UPDATE (Concluded)

(6) Blocks 8 and 9. If the unit has a location or if the unit is a complex nominal unit, transfer control to block L35.

(7) Blocks 10 and 11. There is at least one subordinate unit with no location; set the NOLOC flag equal to one and sum the equipment on hand into a sums array.

(8) Block 12. Increment the no-location subordinate count and place the IUID of the subordination with no location into a list of basic subunits.

(9) Blocks 13 and 14. Call routine GETRCD to retrieve data file 12 trains data for this subordinate and sum into an array of train strengths for basic units. Transfer control to block L33.

(10) Blocks L35 and 15. Set the UPFLAG parameter equal to one for a complex unit, and call routine GETRCD to retrieve data file 12 trains data for this unit.

(11) Block 16. If this unit is not a nominal unit (i.e., basic with location), transfer control to block L778.

(12) Blocks 17 and 18. Sum data file 12 trains data into the nominal trains data array, increment the nominal unit's subordinate count, and place the IUID of the nominal unit into list of subordinate nominal units. Transfer control to block L33.

(13) Block L778. Sum data file 12 trains data into the complex unit trains data array.

(14) Block L33. Sum personnel strength.

(15) Blocks 19 and 20. Call routine GETRCD to retrieve data file 50 authorized personnel strength and add to authorized personnel strength array.

(16) Blocks 21 and 22. Increment subordinate unit counter and load subordinate IUID into subordinate unit list. Also load subordinate UTD into subordinate UTD list.

(17) Block L30. If there are more units in this force to be processed, transfer control to block 4.

(18) Blocks L37 and 23. If there were more than 10 subordinate units, write an error message and return control to the calling routine.

(19) Blocks 24 and 25. A search is made for the superior UTD in the complex UTD table. If it is found, transfer control to block L55.



(20) Blocks 25, 26, and 27. If the superior UTD was not found in the complex part of the UTD table, a search is made to determine if it is a basic UTD. If it is not complex or basic, an error message is printed, and control returns to the calling routine.

(21) Blocks L201 and 28. Call routine GETRCD to retrieve authorized on-hand equipment quantities and sum into unit status record equipment-on-hand array.

(22) Blocks 29 and 30. Call routine GETRCD to retrieve authorized train quantities and sum into basic trains quantity array.

(23) Blocks 31 and 32. Set the NOLOC flag equal to one, write the UTD data to the scratch file and transfer control to block L82.

(24) Blocks L55 and 33. Write the UTD data to the scratch file and call routine GETRCD to retrieve a data file 53 record to check the UTD breakdown of the superior UTD.

(25) Blocks 34 and L82. If the UTDs of the subordinate units do not correspond with those defined in the retrieved data file 53 record, write an informative message.

(26) Blocks 35 and L170. If there is one subordinate unit with no location, set superior unit equipment quantities to contents of equipment-on-hand array, and transfer control to block L181.

(27) Blocks 36, 37, 38, and L150. If the superior unit and all its subordinates have locations write an error message, and zero the superior unit locations. Set the UPFLAG parameter equal to two for a nominal unit.

(28) Block 39. Recompute the BOSS on-hand equipment and trains quantities.

(29) Block L181. Call routine CRAT31 to create data file 31 records for the superior unit.

(30) Blocks 40, 41, and 42. Set the superior unit's status record parameters, and call routine PUTRCD to load the superior unit's status record and data file 50 record.

(31) Block 43. The on-hand equipment quantities of any subordinate unit are changed to percent.

#### 14. ROUTINE CRAT31:

a. Purpose. This routine builds the supply records for vehicles and trains and places them on data file 31.

#### b. Input Variables:

(1) Common ONE and blank common area.

(2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IEOH	Call	Equipment quantities authorized on hand.
IEAUTH	Call	Equipment quantities authorized in trains.
KSTART	Call	Beginning record on data file 31.
ICOMSP	Call	Consumable supplies.
IPERS	Call	Number of personnel authorized.

c. Output. The supply status records are put on data file 31.

d. Logical Flow (Figure II-4-B-10):

(1) Block 1. Zero the working area.

(2) Block 2. Enter into the IVC31 array the quantities consumable supplies authorized on hand and in trains. Begin loop to process all equipment items.

(3) Blocks 3 and 4. If this item is a major item, place the quantity authorized on hand into array IVC31.

(4) Block 6. If this item is not a major item, enter the quantities authorized on hand and in trains into IVC31.

(5) Block 7. ADDRCD places array IVC31 on data file 31.

(6) Blocks 8 and 9. Zero array IVC31. If all equipment items have not been processed, control goes to block 3.

(7) Block 10. Put the number of personnel on data file 31. Return control to the calling routine.

15. ROUTINE RELATR:

a. Purpose. RELATR inserts properly coded indicators of supporting roles into unit status records of a force.

b. Input Variables. Common ONE and blank common area.

c. Output. RELATR outputs the unit status records with appropriate supporting role entries.

d. Logical Flow (Figure II-4-B-11):

(1) Block 1. A do loop is initialized or incremented to pass through all the units of a force.

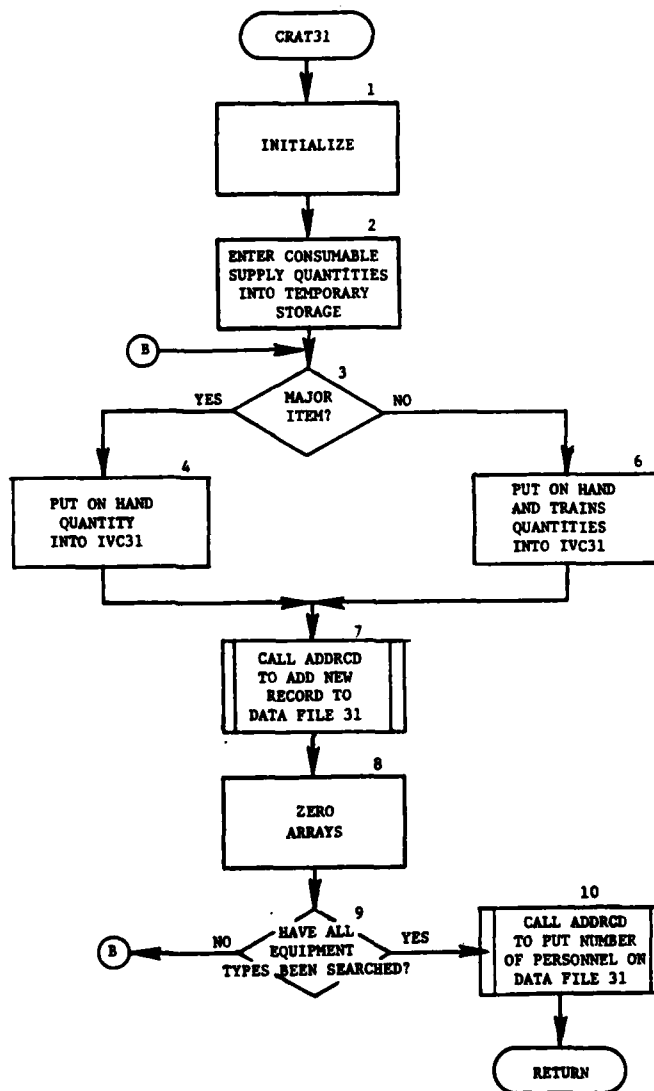


Figure II-4-B-10. Routine CRAT31

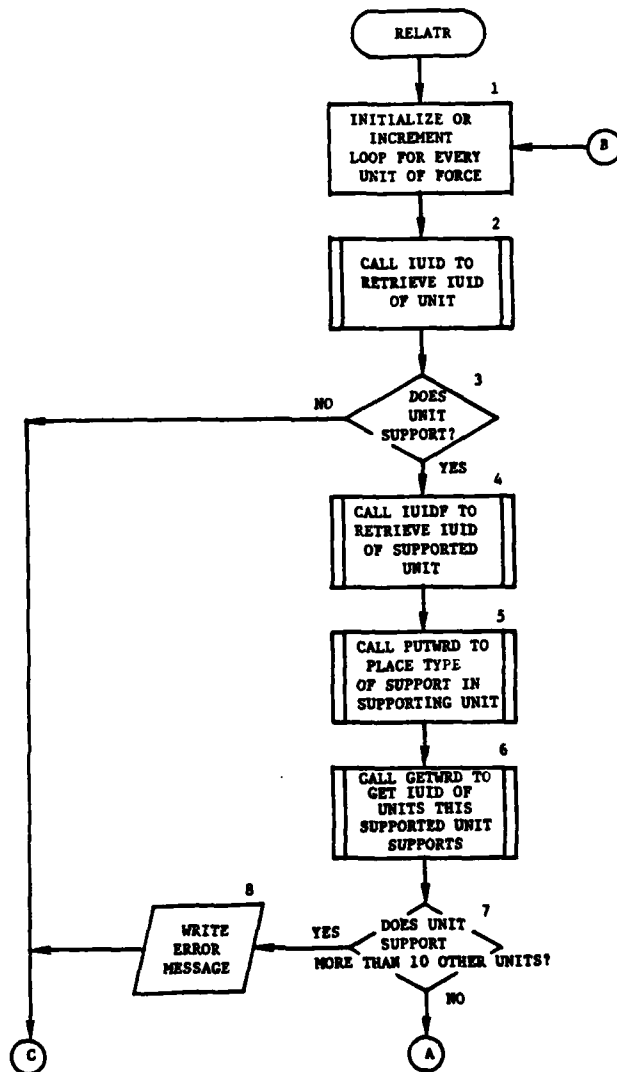


Figure II-4-B-11. Routine RELATR (Continued on Next Page)

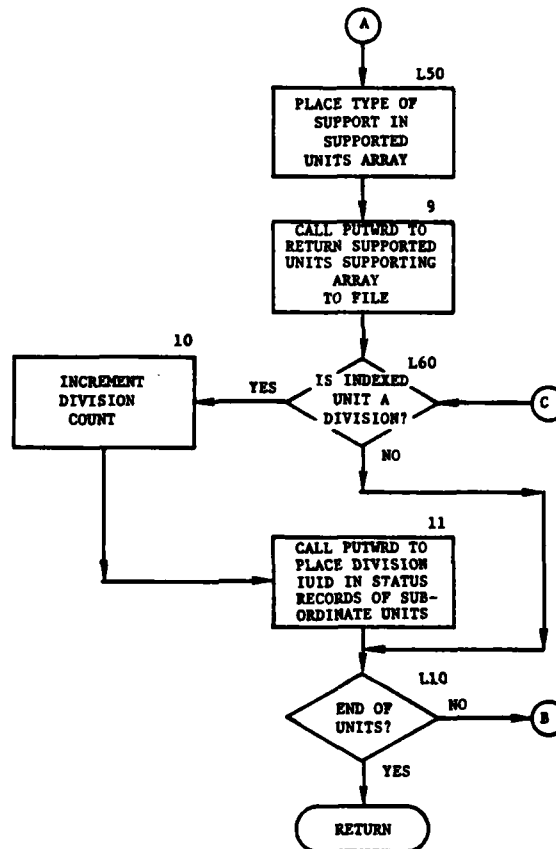


Figure II-4-11. Routine RELATR (Concluded)

(2) Blocks 2 and 3. IUIDF is called to retrieve the IUID of the indexed unit. If that unit is not a supporting unit, transfer control to block L60; otherwise, control proceeds to block 4.

(3) Block 4. IUIDF is called to retrieve the IUID of the supported unit.

(4) Block 5. The type of support is biased by 10000, the IUID of the supported unit is added to it, and a call is made to routine PUTWRD to place this variable in the unit status record of the supporting unit.

(5) Blocks 6, 7 and 8. A call is made to the routine GETWRD to retrieve the IUIDs of any unit that this supported unit in turn supports. If it supports more than 10 units, an error message is written, and control is transferred to block L60.

(6) Blocks L50 and 9. The type of support provided by the original supporting unit is biased by 10000, the IUID of this unit is added to it, and this value is placed in the first vacant location of the supported unit's supporting array. This array is placed in the unit status record of the supported unit.

(7) Blocks L60 and 10. The first and third character of the supporting unit's UTD are extracted. If they are B and M, this unit is a division and the division counter is incremented. If this is not a division, transfer control to block L10.

(8) Block 11. The IUID of the division is placed in the unit status record of every unit subordinate to it.

(9) Block L10. If all units have been processed, a return is made to the calling routine. If there are other units to be processed, transfer control to block 1.

#### 16. ROUTINE SUPPLY:

a. Purpose. SUPPLY builds the supply source portion of a supply status record.

b. Input Variables:

(1) Common ONE. IFNT, NBLUE, and NRED.

(2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ICARD	Card	Card word array.
UID	Card	Unit identification.
ICLASS	Card	Class of supplies (1-10)

<u>Name</u>	<u>Source</u>	<u>Contents</u>
JCARD	Card	Card word array.

c. Output. SUPPLY lists the supply points, units supplied, and classes of supplies. It also updates the appropriate records on data files 1, 31, and 50.

d. Logical Flow. (Figure II-4-B-12):

(1) Bloc's 1 and 2. The first supply data card is read. If it is not a "supply point" card, transfer control to block L900.

(2) Block L10. Call IUIDF with the UID of the supply point to retrieve the IUID of the supply point.

(3) Blocks L15, 3, and 4. Read the next supply data card. If it does not begin with "for units", transfer control to block L100. If it is a "for all units" card, transfer control to block L51.

(4) Blocks 5, 6, and 7. Call IUIDF with the UID of the unit being supplied to retrieve the IUID of that unit. With that IUID, call routine GETWRD to retrieve the status record class I consumption rate for that unit, and sum the consumption into a sum variable.

(5) Blocks 8, 9, and 10. Call routine GETWRD to retrieve the supplied units supplier IUIDs by class, load the new supplier IUID into the appropriate class location and call routine PUTWRD to place the new supply point IUID into the unit's status record.

(6) Block L40. If the last unit on the data card has been processed, transfer control to block L15; otherwise, transfer control to block 5.

(7) Block L100. If the supply point is to supply itself, transfer control to block L102.

(8) Blocks 11, 12, and 13. Call routine GETWRD to retrieve Class I consumption rates for the supply unit from the unit status record (data file 1) and data file 50; update these class I consumption rates by adding the summed class I consumptions. [see paragraph d(5) above]. Call routine PUTWRD to place the new class I consumption rates in the supply point unit's status record and data file 50 record.

(9) Blocks 14, 15, and 16. Call routine GETWRD to retrieve the supply point unit's data file 31 class I consumption record; update these records by adding the summed class I consumption rates [see paragraph d(5) above]. Call routine PUTWRD to place the new consumptions in the supply points data file 31 record.

(10) Block L102. If the last supply data card has been read, return control to the calling routine; otherwise, transfer control to block L15.

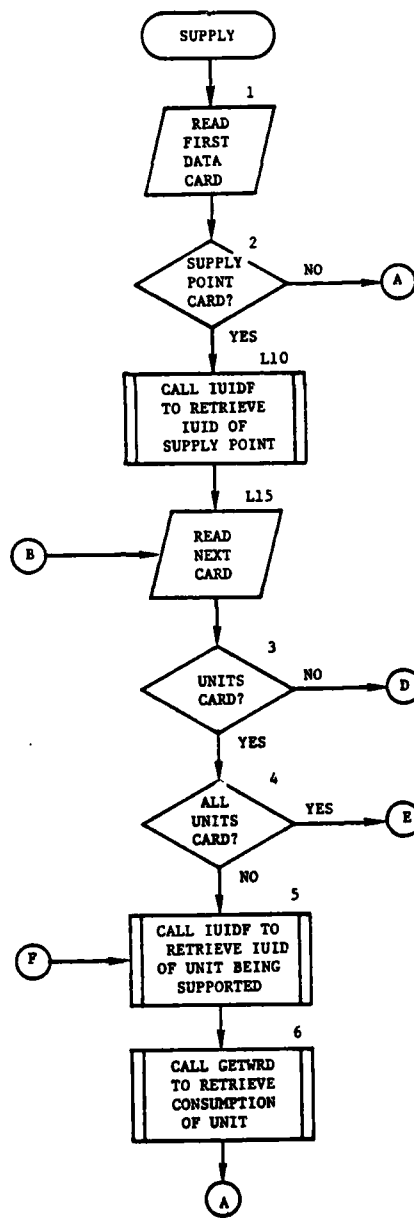


Figure II-4-B-12. Routine SUPPLY (Continued on Next Page)



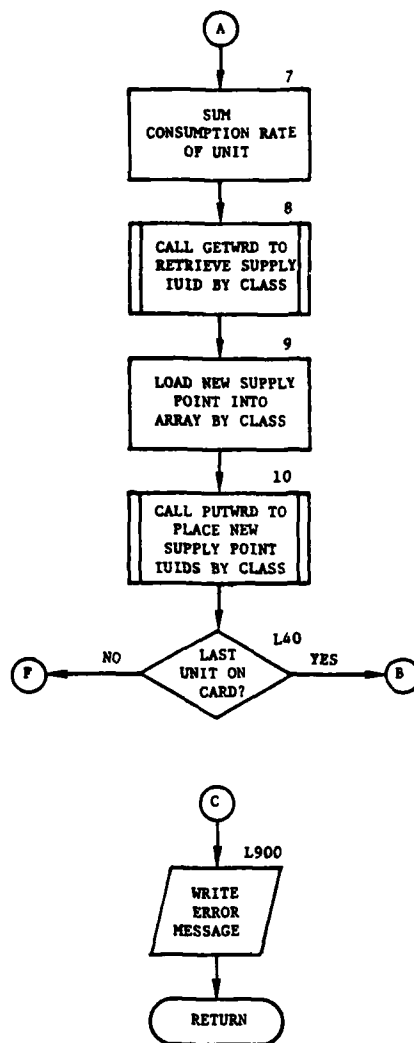


Figure II-4-B-12. Routine SUPPLY (Continued)

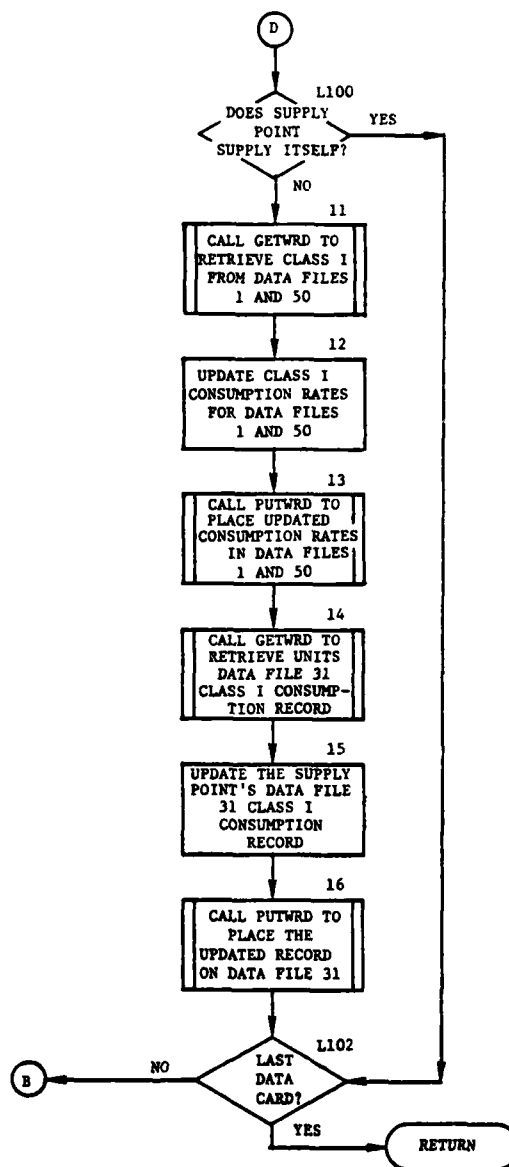


Figure II-4-B-12. Routine SUPPLY (Continued)

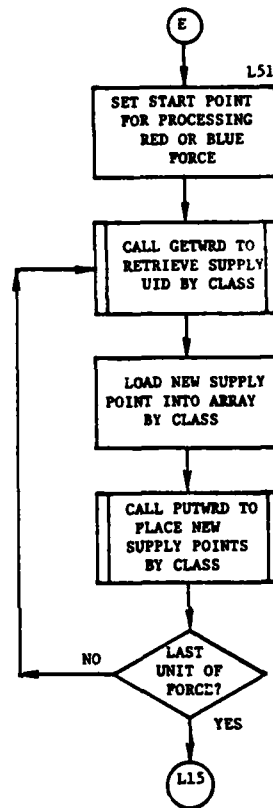


Figure II-4-B-12. Routine SUPPLY (Concluded)

17. ROUTINE TALLY:

a. Purpose. TALLY reads data saved on temporary scratch file (TAPE30), tabulates the UTDs, and checks the task organization requirements.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IUTDCT	TAPE30	Number of UTDs.
UIDD	TAPE30	UID being processed.

c. Output. TALLY checks the task organization requirements and provides informative and destructive messages. It also lists the UTDs and the UIDs from which they were built.

d. Logical Flow (Figure II-4-B-13):

- (1) Block L100. Read a UTD record from the scratch file.
- (2) Block 1. When an end of file is read, control goes to block L400.
- (3) Block 2. Print the UTDs as they are read.
- (4) Block 3. Sum the UTDs and place the sum in temporary storage.
- (5) Block L400. List the UTDs in sequence.
- (6) Blocks 4, 5, and 6. Check the task organization UTD requirements. If a requirement is not met, an error message is printed.
- (7) Block 7. Control goes to block 4 if all UTDs are not checked. If no UTDs remain, control returns to the calling routine.

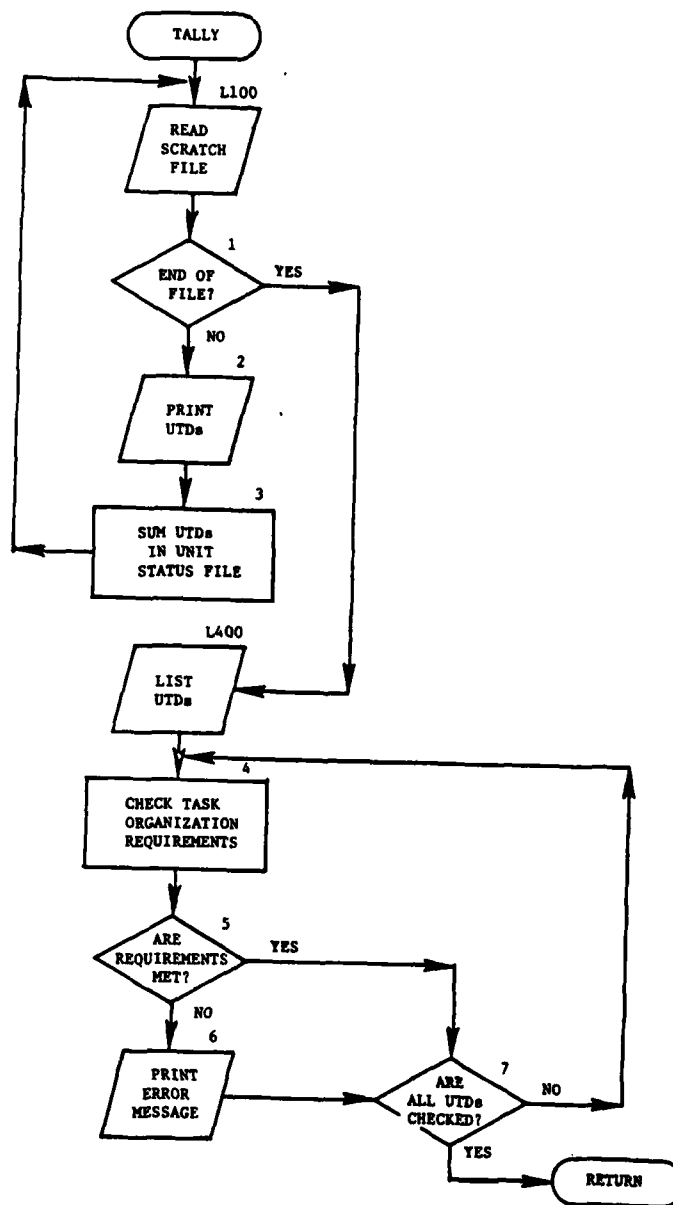


Figure II-4-B-13. Routine TALLY

NOTES

II-4-B-62

## APPENDIX C

### TASK ORGANIZATION DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. Several types of printouts are provided when the ECHFLON routine is executed. These printouts provide descriptions of the unit types available, the task organization, supply point data, unit type designator resolution and usage data, and a list of data file 31 (supply status) records created by the routine.
2. UNIT TYPE DESIGNATOR DIRECTORY. A list of unit type designators available for use within the task organization (see Figure II-4-C-1) is provided for cross reference.
3. CARD IMAGE. A printout of card images of the task organization is provided in Figure II-4-C-2. A description of this printout is contained in Appendix A to Chapter 3, Unit Representation, for the Period Processor.
4. RECOGNI TABLE. As the task organization cards are input, tables are built containing the unit type designators, the unit type designator, X and Y locations, type of support provided, unit being supported, and echelon level of each unit. Figure II-4-C-3 provides a printout of these tables.
5. MESSAGE. An informative message as to the number of divisions in the force is printed as follows:

\*\*\*MESSAGE\*\* THERE WERE 1 DIVISIONS IN THE BLUE FORCE

6. SUPPLY STATUS TABLE. Supply point printout (Figure II-4-C-4) provides a comprehensive listing of the supply status of units within the task organization. The first column contains the unit identification of the supplying unit, the second column contains the unit identification of the unit being supplied, and the third column contains the unit identification record number of the unit being supplied. The next 10 columns contain the unit identification record number of the unit that supplies the corresponding class, as designated by the columnar headings.
7. UNIT RESOLUTION LIST. If a basic unit within the task organization is a complex type, it must be resolved into a basic type unit. In Figure II-4-C-5, lines one and two, states that unit B5502BEN was placed in the task organization as a basic unit, but was of a complex type. It was thus resolved into the basic structure of one type KCSE, three types RCSE, and one type RESE.

AMLL	APLL	AMCC	APLC	ATCC	KCC	KCCI	QNTA	KAMI	KLSI
KICT	KIMT	KCST	JACA	ICOT	JOLT	ICOT	JOLI	IAGS	JALG
ITCT	ILLT	ICST	PAFA	PMHT	PMCT	IMCT	ILLI	ILSI	PNFA
PISI	PIMI	PICI	IACI	ISLI	IONT	IFSI	IBCI	IJLI	ICMI
IMSI	IACH	IKMH	VALH	VCLH	IRCH	IRLH	IRMH	JACH	JASH
JBSH	JCSH	VELH	VALH	IFCA	IFFA	IFFA	IFLA	IGCA	IGFA
IGLA	PACA	PEFA	KCCA	KGFA	KSLA	KSCA	KHFA	KALA	KFCA
KCFA	KCLA	KACA	KLFA	KLLA	JHCD	JAFD	JAFD	JECE	RASF
PCSE	PESE	QCSE	QSEF	PASE	QBSE	ESCS	JSCS	PBCS	IMCC
IMLC	IMHC	IMMH	PMFH	PSMH	IJCJ	JITJ	UATJ	U9IJ	UDIJ
UHIJ	UCIJ	JSIJ	JGII	USIJ	JNCL	JLLI	JTCL	JSLL	PSLL
JTLL	PTLL	JMCL	JHLL	JFLL	JALL	POLL	JSCL	JPLL	JMLL
JNCL	JNLL	JOLL	KHCT	KHFD	KACH	KASH	KBSH	KALH	KOLE
KLLE	PRBE	QURE	KHCE	RCSE	RESE	VFIJ	VITI	UFIJ	ICIJ
ABLL	ABLL	HATT	NQII	DRRY	ATTY	LFOT	LFIA	DRLT	GINC
HESE	DSTJ	GVTJ	LECE	GFFA	LBCT	LINC	SCSE	GTLT	NFFD
NEFD	GEMT	GEMT	LBCI	GLLI	GRMT	MAFA	NGFD	MBFA	DACF
GPSE	GRSE	DYSE	WDSF	DTCA	GHFA	MGFA	GRFA	DMCA	MMFA
DMCA	MPFA	GEFA	DMCD	MAFD	NLCA	NLFA	NHCA	NACD	NDFD
GECE	MDSE	MOSE	DMCD	MBFD	NDSE	SOSE	MBSE	SBSE	NTSF
SBXE	NOME	SNWE							

Figure II-4-C-1. Unit Type Designator Directory, Task Organization



OPERATIONS BLUE UNIT.  
 TASK ORGANIZATION  
 B1000ZFC=\*TOE AARB.  
 B1000DIV=\*TOE RAMC.  
 B1AFCMR2 \*TOE AFFY \*LOCATION 0070000-0044000.  
 B101A00P \*TOE BTCC \*LOCATION 0090100-0098100.  
 B1DIVZTP=\*TOE C00C.  
 B1MAINCP \*TOE BMCC \*LOCATION 0090100-0098100.  
 B10CRFCP \*TOE BRLO \*LOCATION 0090100-0098100.  
 B1000ZEN=\*TOE E00E \*LOCN 0091100-0095100.  
 B101PREN \*TOE URSE.  
 B101HOEN \*TOE DECE.  
 B101OMEN \*TOE RASE.  
 END OF B1000ZEN.  
 B1022ZAI=\*TOE E0MI \*LOCATION 0087300-0095500.  
 B122HQAI=\*TOE IACI.  
 B140RNTI \*TOE URIJ.  
 END OF B122HQAI.  
 B1022AAI \*TOE IBMI.  
 B1022RAI \*TOE IBMI.  
 B1022CAI \*TOE IBMI.  
 B122CSAI \*TOE IFSI.  
 END OF B1022ZAI.  
 B1000ZSC=\*TOE ESSS.  
 B101SC0N \*TOE ESCS \*LOCATION 0090100-0098100.  
 END OF B1000ZSC.  
 B1000ZAD=\*TOE CASD.  
 B100H0AD \*TOE JHCR \*LOCATION 0089100-0094100.  
 END OF B1000ZAD.  
 B1000ZTI=\*TOE EIIJ \*LOCATION 0089500-0098300.  
 B100H0TI \*TOE IJCJ.  
 B100SPTI=\*TOE JIIJ.  
 B130RNTI \*TOE J4IJ.  
 END OF B100SPTI.  
 B100TSTI \*TOE JSIJ.  
 END OF B1000ZTI.  
 END OF B100IV7TP.

Figure II-4-C-2. Card Image, Task Organization

\*\*INPUT-IN-RECOGN1 TABLES

N	UID	UTC	X	Y	NAUP	NSUP	LEV
1	B1000ZFC	AAFB			0		1
2	B1000DIV	BAMC			0		2
3	B1AFBMR	AFBY	0070000	0044000	0		3
4	B10TACCP	RTCC	0000100	0098100	0		3
5	B10DIV7TP	CDCC			0		3
6	B1MAI JOP	RMCC	0030100	0098100	0		4
7	B100PRCP	RRCC	0000100	0098100	0		4
8	B10007EN	EESE	0001100	0095100	0		4
9	B1013REN	JBSE			0		5
10	B101H0EN	JESE			0		5
11	B10A0MEN	PASE			0		5
12	B10227AI	ECMI	0037300	0095500	0		4
13	B122H0AI	IACI			0		5
14	B140BNTI	UBIJ			0		6
15	B1022AAI	IBMI			0		5
16	B1022RAI	IBMI			0		5
17	B1022CAI	IBMI			0		5
18	B1220SAI	IFSI			0		5
19	B1000ZSC	ESSS			0		4
20	B10150BN	EDCS	0090100	0098100	0		5
21	B1000ZAD	EASD			0		4
22	B1000H0A	JH00	0039100	0094100	0		5
23	B1000ZTI	EITJ	0039600	0098800	0		4
24	B1000H0T	IJ0J			0		5
25	B1000SPTI	JITJ			0		5
26	B12ARNTI	UH0J			0		5
27	B100TSTI	JSIJ			0		5
28	B10SPTZMD	CSCL	0030100	0101300	0		3
29	B1400SCM	J0CL			0		4
30	B1000MIN	JLLL			0		4
31	B10150SP	JSOS			0		4
32	B10007MN	FMLL			0		4
33	B1400TMN	JMCL			0		5
34	B1000VYMN	JHLL			0		5
35	B1000ACMN	JALL			0		5
36	B10007ST	ESLL	0036000	0107000	0		4
37	B1000HQST	JTCL			0		5
38	B1000SSST	JSLL	0036000	0107000	0		5
39	B1000TRST	JTLL			0		5
40	B1000ZMC	ENLL			0		4
41	B1000HMC	JNCL			0		5
42	B1000VZDA	CAFA			0		3
43	B1000H0FA	JACA	0032200	0090800	0		4
44	B1000FATI	UDIJ			0		5
45	B5015ZGS	EGFA	0108200	0087800	0		4
46	B5015H0GS	K0CA			0		5
47	B5015AGS	KGFA			0		5
48	B50150GS	KGFA			0		5
49	B50150GS	KGFA			0		5

Figure II-4-C-3. RECOGN1 Table, Task Organization

SUPPLY POINT	UNIT	SUPPLIER	UID	CLASS01	CLASS02	CLASS03	CLASS04	CLASS05	CLASS06	CLASS07	CLASS08	CLASS09	CLASS10
B100M2LL	54	B100M2LL	54	0	0	0	0	279	0	0	0	0	0
B100M2LL	286	B100M2LL	286	0	0	0	0	279	0	0	0	0	0
B100M2LL	66	B100M2LL	66	0	0	0	0	279	0	0	0	0	0
B100M2LL	257	B100M2LL	257	0	0	0	0	279	0	0	0	0	0
B100M2LL	259	B100M2LL	259	0	0	0	0	279	0	0	0	0	0
B100M2LL	258	B100M2LL	258	0	0	0	0	279	0	0	0	0	0
B100M2LL	176	B100M2LL	176	0	0	0	0	279	0	0	0	0	0
B100M2LL	177	B100M2LL	177	0	0	0	0	279	0	0	0	0	0
B100M2LL	178	B100M2LL	178	0	0	0	0	279	0	0	0	0	0
B100M2LL	179	B100M2LL	179	0	0	0	0	279	0	0	0	0	0
B100M2LL	180	B100M2LL	180	0	0	0	0	279	0	0	0	0	0
B100M2LL	251	B100M2LL	251	0	0	0	0	279	0	0	0	0	0
B100M2LL	283	B100M2LL	283	0	0	0	0	279	0	0	0	0	0
B100M2LL	252	B100M2LL	252	0	0	0	0	279	0	0	0	0	0
B100M2LL	45	B100M2LL	45	0	0	0	0	279	0	0	0	0	0
B100M2LL	175	B100M2LL	175	0	0	0	0	279	0	0	0	0	0
B100M2LL	29	B100M2LL	29	0	0	0	0	279	0	0	0	0	0
B100M2LL	54	B100M2LL	54	0	0	0	0	279	0	0	0	0	0
B100M2LL	56	B100M2LL	56	0	0	0	0	279	0	0	0	0	0
B100M2LL	17	B100M2LL	17	0	0	0	0	279	0	0	0	0	0
B100M2LL	44	B100M2LL	44	0	0	0	0	279	0	0	0	0	0
B100M2LL	57	B100M2LL	57	0	0	0	0	279	0	0	0	0	0
B100M2LL	253	B100M2LL	253	0	0	0	0	279	0	0	0	0	0
B100M2LL	70	B100M2LL	70	0	0	0	0	279	0	0	0	0	0
B100M2LL	72	B100M2LL	72	0	0	0	0	279	0	0	0	0	0
B100M2LL	78	B100M2LL	78	0	0	0	0	279	0	0	0	0	0
B100M2LL	90	B100M2LL	90	0	0	0	0	279	0	0	0	0	0
B100M2LL	84	B100M2LL	84	0	0	0	0	279	0	0	0	0	0
B100M2LL	95	B100M2LL	95	0	0	0	0	279	0	0	0	0	0
B100M2LL	142	B100M2LL	142	0	0	0	0	279	0	0	0	0	0
B100M2LL	174	B100M2LL	174	0	0	0	0	279	0	0	0	0	0
B100M2LL	181	B100M2LL	181	0	0	0	0	279	0	0	0	0	0
B100M2LL	202	B100M2LL	202	0	0	0	0	279	0	0	0	0	0
B100M2LL	234	B100M2LL	234	0	0	0	0	279	0	0	0	0	0
B100M2LL	242	B100M2LL	242	0	0	0	0	279	0	0	0	0	0
B100M2LL	249	B100M2LL	249	0	0	0	0	279	0	0	0	0	0
B100M2LL	248	B100M2LL	248	0	0	0	0	279	0	0	0	0	0
B100M2LL	241	B100M2LL	241	0	0	0	0	279	0	0	0	0	0
B100M2LL	240	B100M2LL	240	0	0	0	0	279	0	0	0	0	0
B100M2LL	55	B100M2LL	55	0	0	0	0	279	0	0	0	0	0

Figure II-4-C-4. Supply Status Table, Task Organization

UID B5512AFN WAS BUILT FROM THE FOLLOWING UIDS  
KOSE 1 ROSE 3 PESE 1

UID B5512CFN WAS BUILT FROM THE FOLLOWING UIDS  
KOSE 1 ROSE 3 PESE 1

UID B5512DFN WAS BUILT FROM THE FOLLOWING UIDS  
KOSE 1 ROSE 3 PESE 1

UID B5512MAB WAS BUILT FROM THE FOLLOWING UIDS  
KOSE 1 ROSE 3 PESE 1

UID B30372GS WAS BUILT FROM THE FOLLOWING UIDS  
FSFA 1 KFOA 1 KOFA 3 KOLA 1

UID B30374GS WAS BUILT FROM THE FOLLOWING UIDS  
FHFA 1 KFOA 1 KHFA 3 KALA 1

UID B1AFRMDR WAS BUILT FROM THE FOLLOWING UIDS  
AFFY 1 VFIJ 6 VTTY 1

UID B541RSUR WAS BUILT FROM THE FOLLOWING UIDS  
KOSH 1 UFIJ 12 TCIJ 1

UID PA1X1AFN WAS BUILT FROM THE FOLLOWING UIDS  
MASE 1 NOSE 1 SDSE 2

UID RA1X1BFN WAS BUILT FROM THE FOLLOWING UIDS  
MASE 1 NOSE 1 SDSE 2

UID RA1X1CFN WAS BUILT FROM THE FOLLOWING UIDS  
MASE 1 NOSE 1 SDSE 2

UID RA1X2AFN WAS BUILT FROM THE FOLLOWING UIDS  
MASE 1 NOSE 1 SDSE 2

UID RA1X2BFN WAS BUILT FROM THE FOLLOWING UIDS  
MASE 1 NOSE 1 SDSE 2

Figure II-4-C-5. Task Organization Unit Resolution List

II-4-C-6

8. UNIT TYPE DESIGNATOR SUMMARY. Figure II-4-C-6 shows a list summarizing all unit types and the number of times each was encountered within the task organization routine.

9. UNITS CREATED. Figure II-4-C-7 shows a list of all the supply records (data file 31) created within the task organization routine and the associated units.

THE FOLLOWING IS A SUMMARY OF UTOS USED IN SEQUEN

UBIJ 8	UNIJ 3	JAFD 2	JBFN 2	UCIJ 1	IMLC 1	VB-LH 1	VFLM 2	VCL4 1	VALM 3
IRLH 1	JSCL 1	JPLL 1	JHLL 1	JBSE 1	PSSE 2	GRSE 1	JECE 1	RASE 1	IACI 2
IBMI 6	IFSI 2	FSCS 1	JHCG 1	IJCJ 1	JLIJ 1	JSIJ 1	JHCL 1	JHLL 1	JALL 1
JTCL 1	JSLL 1	JTLL 1	JNCL 1	UOIJ 1	KCCA 1	KGFA 3	KSLA 1	KGCA 2	KHFA 6
KALA 2	KFCA 3	KGFA 9	KCLA 3	KACA 3	KLFA 2	KLLA 1	UAIJ 3	UGIJ 6	USTJ 6
PRCS 3	JCSF 2	PCSE 6	PESE 2	IFCA 1	IFFA 1	IBFA 1	IFLA 1	ITCI 2	ITMT 6
PHMT 14	PMCT 6	IAMI 3	PIPI 9	PICI 3	IUST 2	PMFA 2	IMCI 1	ILSI 1	PNFA 1
PTST 2	THCC 1	IHMC 2	IAMH 1	PSMH 1	PMFH 1	JNLL 2	PSLL 3	PTLL 3	JFLL 1
ILIT 2	ICSE 1	ICSE 6	QCESE 2	IGCA 1	IGFA 3	ICFA 1	PACA 1	PFFA 1	IGLA 1
IRCI 1	ICMT 3	IMSI 1	JACH 1	JASH 1	JBSH 3	JCSH 2	JCLH 1	JOLL 1	POLL 1
ISLI 2	IACH 1	IKMH 3	IALH 1	IGHI 1	IAMH 3	FALL 1	KHCI 2	KAMI 6	KMT 6
KLST 2	ONFA 4	KTCY 2	KOST 2	KHCE 2	KOSE 8	RCSE 2	RESF 9	KLLE 1	KISE 2
PBRE 6	OBEE 2	KOLE 1	BMCC 1	ERLC 1	FESE 1	ECMI 2	ESSS 1	EASJ 1	EIJ 1
JNCL 1	JULL 1	JSCS 1	FHLL 1	ESLL 1	FHLL 1	JACA 1	FSCA 1	FMEA 2	FSEA 3
FMEA 1	ICCI 1	EFCA 1	ERMT 2	EGMI 1	ECMC 1	JCLY 1	IDCI 1	EGFA 1	EDMI 1
EDSH 1	JDLI 1	IACC 1	EAMH 1	ERHM 1	JALC 1	KCCT 1	FMMI 2	FMFT 2	FESC 2
KHCD 1	KHFD 3	KASH 3	KASH 1	KACH 1	KALH 1	AFHY 1	VFIJ 5	VTFY 1	BTCC 1
CNCC 1	CSCI 1	CAFA 1	COYT 1	CDMI 1	CANC 1	CHMI 1	CZSF 1	FHFD 1	EASH 1
KCSH 1	UFIJ 12	ICIJ 1	AMLL 1	APLL 1	RAMC 1	NHCA 3	ABAB 1	GECE 4	MASE 18
NDSE 19	SDSF 57	MOSE 4	MCSE 4	NLCA 3	NLFA 9	MGFA 3	NACD 3	NDFJ 12	LINC 12
SCSE 9	NFFD 9	NEFD 12	MPFA 9	DTCA 3	MGFA 12	LECE 3	MGFA 3	NRSF 3	MAFA 6
DNCD 2	MMFD 5	LFIA 3	GMHC 3	ORLT 3	4ASE 4	GLFA 3	MMSE 3	DMCA 2	SASE 6
MTSE 7	MTSE 7	S9XE 7	MMSE 11	NRWE 11	SMHE 22	LECE 3	GLFA 3	DGFA 3	DDFD 3
LACT 9	GTLT 9	GEMI 27	GEMI 9	MAFA 3	MPFA 2	IBCI 3	NGFD 3	G.LI 3	GRMT 3
GFPA 1	DACE 2	GPSE 2	GRSE 2	DYSE 2	MOSE 2	DMCA 1	DNCN 1	MAFJ 3	OSTJ 1
GVIJ 1	HATT 2	NRH 1	DLFA 3	DMFA 3	DMFJ 2	LFCT 3	GASE 3	DDFA 3	DTMT 9
DTMI 3	ATTY 4	DESE 2	DYSE 2	HESE 2	DMFA 1	DMFD 1	DTTJ 1	BEFA 1	DRRY 6
DRMT 3	DFSE 1	BAFA 1	ACCY 1	AALL 1	ABLL 1	AARR 1			

Figure II-4-C-6. Unit Type Designator Summary, Task Organization

TABLE II-4-C-7. Task Organization Units Created.

UNIT ORGANIZATION									
UNIT	1	2	3	4	5	6	7	8	9
UIOZ 0101000	1	1328	1328	1328	1328	1328	1328	1328	1328
2	1328	1328	1328	1328	1328	1328	1328	1328	1328
3	10000	10000	10000	10000	10000	10000	10000	10000	10000
7	1000	1000	1000	1000	1000	1000	1000	1000	1000
9C	100	100	100	100	100	100	100	100	100
96	10	10	10	10	10	10	10	10	10
97	10	10	10	10	10	10	10	10	10
152	1	1	1	1	1	1	1	1	1
167	1	1	1	1	1	1	1	1	1
201	5	5	5	5	5	5	5	5	5
UIOZ 0101200	1	35	35	35	35	35	35	35	35
2	1328	1328	1328	1328	1328	1328	1328	1328	1328
3	10000	10000	10000	10000	10000	10000	10000	10000	10000
7	1000	1000	1000	1000	1000	1000	1000	1000	1000
9C	100	100	100	100	100	100	100	100	100
96	10	10	10	10	10	10	10	10	10
97	10	10	10	10	10	10	10	10	10
162	1	1	1	1	1	1	1	1	1
167	1	1	1	1	1	1	1	1	1
201	5	5	5	5	5	5	5	5	5
UIOZ 0101400	1	35	35	35	35	35	35	35	35
2	1328	1328	1328	1328	1328	1328	1328	1328	1328
3	10000	10000	10000	10000	10000	10000	10000	10000	10000
7	1000	1000	1000	1000	1000	1000	1000	1000	1000
9C	100	100	100	100	100	100	100	100	100
96	10	10	10	10	10	10	10	10	10
97	10	10	10	10	10	10	10	10	10
162	1	1	1	1	1	1	1	1	1
167	1	1	1	1	1	1	1	1	1
201	5	5	5	5	5	5	5	5	5
UIOZ 0101600	1	35	35	35	35	35	35	35	35
2	1328	1328	1328	1328	1328	1328	1328	1328	1328
3	10000	10000	10000	10000	10000	10000	10000	10000	10000
7	1000	1000	1000	1000	1000	1000	1000	1000	1000
9C	100	100	100	100	100	100	100	100	100
96	10	10	10	10	10	10	10	10	10
97	10	10	10	10	10	10	10	10	10
162	1	1	1	1	1	1	1	1	1
167	1	1	1	1	1	1	1	1	1
201	5	5	5	5	5	5	5	5	5
UIOZ 0101800	1	35	35	35	35	35	35	35	35
2	1328	1328	1328	1328	1328	1328	1328	1328	1328
3	10000	10000	10000	10000	10000	10000	10000	10000	10000
7	1000	1000	1000	1000	1000	1000	1000	1000	1000
9C	100	100	100	100	100	100	100	100	100
96	10	10	10	10	10	10	10	10	10
97	10	10	10	10	10	10	10	10	10
162	1	1	1	1	1	1	1	1	1
167	1	1	1	1	1	1	1	1	1
201	5	5	5	5	5	5	5	5	5

Figure II-4-C-7. Task Organization Units Created

NOTES

II-4-C-10



APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR TASK ORGANIZATION

(AVAILABLE UNDER SEPARATE COVER)

NOTES

II-4-D-2

## CHAPTER 5

### ENVIRONMENTAL DATA INPUT PROGRAMS

1. INTRODUCTION. The Environmental data input programs create and load the constant data files describing the environmental representation. Those data describe the terrain characteristics and elevation and the weather conditions.

2. ROUTINES:

a. General. The Environmental data input programs consist of three separate and independent programs as described below.

b. Elevation. Routine ELEVLD creates and loads the terrain elevation data file and produces a dump of the file contents. It requires a digitized terrain tape as input and determines and stores the elevation of each point on the elevation grid.

c. Terrain. Routine TERNLD reads the terrain characteristic data from the input cards and creates and loads the terrain characteristics file. Routine TERNDP produces a formatted dump of the file contents.

d. Weather. The weather data are read from cards and loaded into the weather data file by routine WETHLD. That routine also produces a formatted dump of the contents of the weather file.

3. FILES. The following data files are created and loaded.

- . Terrain elevation file            - data file 13
- . Terrain characteristics file - data file 3
- . Weather file                    - data file 4

NOTES

II-5-2

## APPENDIX A

### ENVIRONMENTAL DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for environment are documented in Appendix A, Environmental Representation Input Requirements, to Chapter 4 of the Period Processor (Section IV).

NOTES

II-5-A-2

## APPENDIX B

### ENVIRONMENTAL DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. The environmental representation data are loaded by three independently operated routines. Routine ELEVLD loads the terrain elevation data. This differs from all other DIVWAG load routines in that the input medium is magnetic tape rather than cards. The routine TERAIn assisted by TERNLD and TERNOP routines, loads the terrain characteristic data other than elevation, and the routine WETHLD loads the weather data. These routines are described in this appendix.

#### 2. ROUTINE ELEVLD:

a. Purpose. Routine ELEVLD creates and loads the terrain elevation data (data file 13). It is designed to extract the elevation data from a digitized terrain tape. The tape used for all applications to date has been obtained from the TACOS Model data base.

b. Input Variables. The only source of input data for this routine is the digitized terrain tape. The tape is structured by 100,000-meter map squares with 24 logical records per square. The four squares used for recent studies are:

- (1) MA : Records 122 through 145
- (2) MB : Records 218 through 241
- (3) NA : Records 146 through 169
- (4) NB : Records 242 through 265

Each record contains the elevations of 1717 points spaced at 500-meter intervals (101 points in the easterly direction by 17 points in the northerly direction), except those describing the northern boundary of the squares that have only 14 points in the northerly direction. Each elevation point occupies 16 bits on the tape and is in binary integer notation in decameters. The layout of the records on the tape as they relate to the terrain areas is depicted in Figure II-5-B-1.

c. Output Variables. The output from the routine is data file 13. That data file contains one word for each elevation point. Each record contains the elevation in meters of all points on a given east-west row. There is a record for each row. For recent projects, the file was composed of 401 records of 401 words each to describe a terrain area 200,000 by 200,000 meters with 500-meter elevation grid intervals.

#### d. Logical Flow (Figure II-5-B-2):

(1) Blocks 1 and 1A. Bring in the file name table. If data file 13 has been created and is of the correct size, transfer control to block L100.

MB	NB
MA	NA

MB - 23 RECORD 240	MB - 24 RECORD 241	NB - 23 RECORD 264	NB - 24 RECORD 265
MB - 21 RECORD 238	MB - 22 RECORD 239	NB - 21 RECORD 262	NB - 22 RECORD 263
MB - 5 RECORD 222	MB - 6 RECORD 223	NB - 5 RECORD 246	NB - 6 RECORD 247
MB - 3 RECORD 220	MB - 4 RECORD 221	NB - 3 RECORD 244	NB - 4 RECORD 245
MB - 1 RECORD 218	MB - 2 RECORD 219	NB - 1 RECORD 242	NB - 2 RECORD 243
MA - 23 RECORD 144	MA - 24 RECORD 145	NA - 23 RECORD 168	NA - 24 RECORD 169
MA - 21 RECORD 142	MA - 22 RECORD 143	NA - 21 RECORD 166	NA - 22 RECORD 167
MA - 7 RECORD 128	MA - 8 RECORD 129	NA - 7 RECORD 152	NA - 8 RECORD 153
MA - 5 RECORD 126	MA - 6 RECORD 127	NA - 5 RECORD 150	NA - 6 RECORD 151
MA - 3 RECORD 124	MA - 4 RECORD 125	NA - 3 RECORD 148	NA - 4 RECORD 149
MA - 1 RECORD 122	MA - 2 RECORD 123	NA - 1 RECORD 146	NA - 2 RECORD 147

Figure II-5-B-1. Record Layout of Terrain Elevation Tape

II-5-B-2



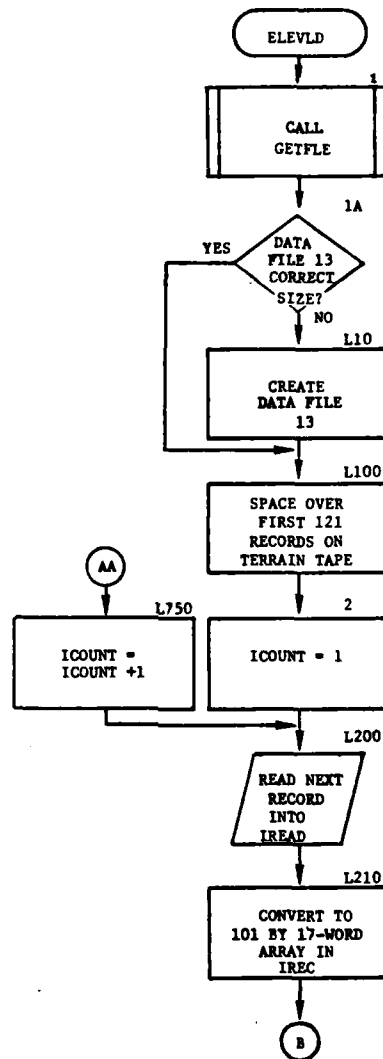


Figure II-5-B-2. Routine ELEVLD (Continued on Next Page)

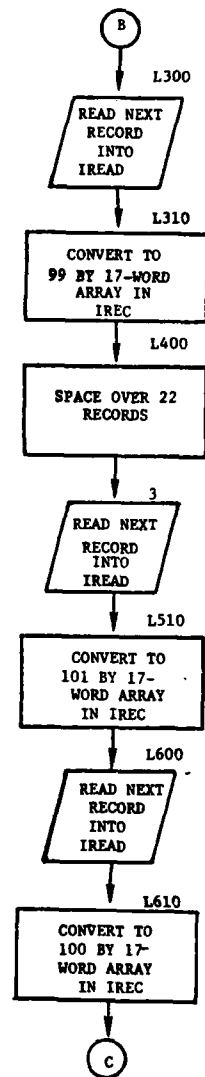


Figure II-5-B-2. Routine ELEVL D (Continued)

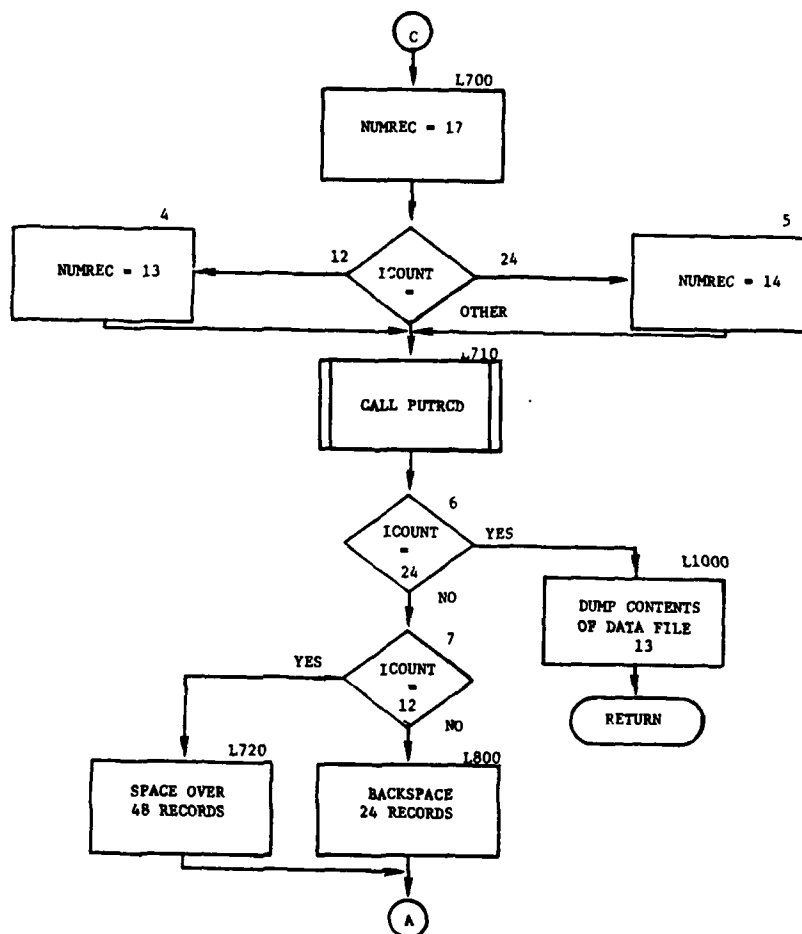


Figure II-5-B-2. Routine ELEVLD (Concluded)

(2) Block L10. If the file has been previously created, it is removed. The file is then created with the required dimensions.

(3) Block L100. The tape is positioned to record 122 to begin reading the elevation of square MA.

(4) Block 2. ICOUNT is initialized to one. It counts the number of four-record sets as they are read and processed.

(5) Block L200. A tape record is read into IREAD array. This contains the 1717 elevation points describing the westernmost of the four blocks.

(6) Block L210. Each 16-bit word is taken in sequence from IREAD, converted to an integer word, multiplied by 10, and stored in IREC. This process fills the first 101 columns in the IREC array.

(7) Block L300. The next tape record is read into IREAD. This contains the 1717 elevation points describing the next block to the east.

(8) Block L310. The process described in block L210 (paragraph d(6) above) is performed to fill columns 102 through 200 of the IREC array.

(9) Block L400. The tape is positioned to read the third record of this set by spacing over 22 records.

(10) Block 3. A tape record is read into IREAD. This contains the 1717 elevation points describing the next block to the east.

(11) Block L510. The process described in block L210 is performed to fill columns 201 through 301 of the IREC array.

(12) Block L600. The next record is read into IREAD. This contains the 1717 elevation points describing the easternmost block of the set.

(13) Block L610. The process described in block L210 is performed to fill columns 302 through 401 of the IREC array.

(14) Block L700. IREC is normally filled with 17 rows and 401 columns that correspond to 17 records on data file 13.

(15) Blocks 4 and 5. If this set of records pertains to the northernmost boundary of NA and MA, only 13 records are included. If it pertains to the northernmost boundary of NB and MB, only 14 records are included.

(16) Block L710. The correct number of records is put onto data file 13 from IREC.

(17) Block 6. If all 24 sets have been processed, transfer control to block L1000; otherwise, transfer control to block 7.

(18) Blocks 7 and L720. If ICOUNT is equal to 12, the tape is spaced forward 48 records to position it to read the first record of square MB, and control is transferred to block L750.

(19) Block L800. If ICOUNT is not equal to 12, the tape is backspaced 24 records to read the first record of the next set.

(20) Block L750. ICOUNT is incremented by one to begin processing the next set, and control is transferred to block L200.

(21) Block L1000. The contents of data file 13 are printed.

3. ROUTINE TERAIRN. TERAIRN is the controlling program for loading the terrain data deck. It calls the TERNLD routine to read the data cards and build the terrain file (data file 3). It also calls the TERNDP routine to print the newly created terrain file (data file 3).

4. ROUTINE TERNLD:

a. Purpose. The TERNLD routine reads the terrain data cards, performs minor editing, and writes the acceptable data onto the terrain file (data file 3).

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
NX	Card	Number of cells in the X direction.
NY	Card	Number of cells in the Y direction.
NTERR	Card	Size of a terrain cell.
ISTORE	Card	RVSS codes.
IX	Card	X coordinate for ISTORE.
IY	Card	Y coordinate for ISTORE.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
ITERR	DF3	100 terrain records.

d. Logical Flow (Figure II-5-B-3):

(1) Block L100. Read the first card; it is the parameter card specifying the number of X coordinates, Y coordinates, and size of the cell.

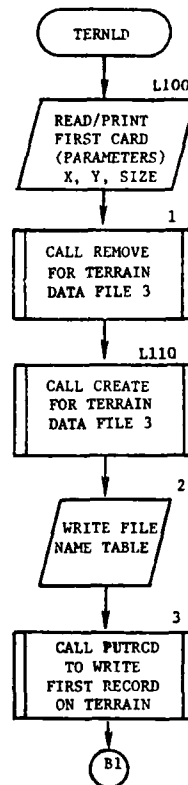


Figure II-5-B-3. Routine TERNLD (Continued on Next Page)

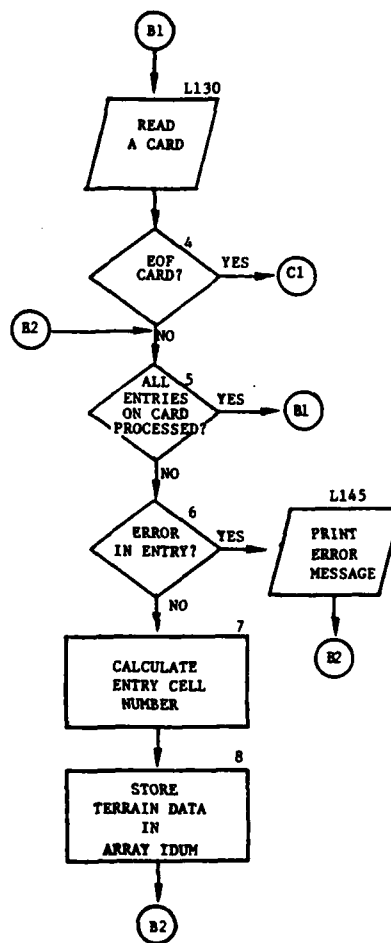


Figure II-5-B-3. Routine TERNLD (Continued)

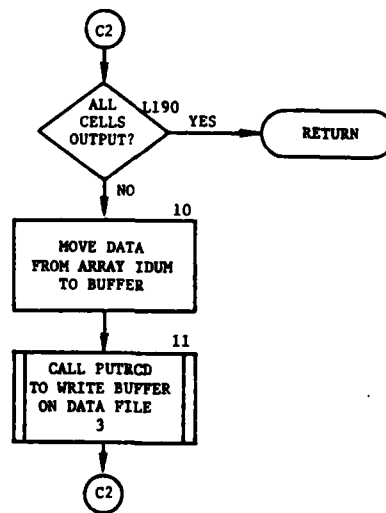
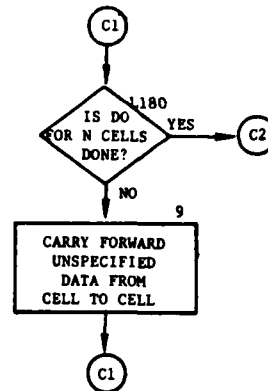


Figure II-5-B-3. Routine TERNLD (Concluded)



(2) Blocks 1, L110, and 2. Remove the previous data file 3, if one exists, and create a new data file 3 of the required size. Print the new file name table.

(3) Block 3. Build the first record for data file 3 from the parameter card and put it on data file 3 by calling PUTRCD.

(4) Block L130. Read the next data card.

(5) Block 4. If an end of file was detected, or if a data end of file card was read, control goes to block L180.

(6) Block 5. If all entries on the card have been stored in array IDUM, return to block L130. If not, increase entry pointer by one.

(7) Blocks 6 and L145. If the entry has an edit error, print an error message and return to block 5.

(8) Block 7. Calculate the entry cell number.

(9) Block 8. Store the terrain data in array IDUM.

(10) Blocks L180 and 9. Examine each cell's data in array IDUM, and set any unspecified data item equal to that data in the previous cell.

(11) Blocks L190, 10, and 11. Move the cell data from array IDUM, and build a buffer of 100 records. When the buffer is full or there are no more entries in IDUM, output the buffer to data file 3 by calling PUTRCD. When all records are output return control to the calling routine.

#### 5. ROUTINE TERNDP:

a. Purpose. The TERNDP routine is called by the TERAIn routine, and lists the terrain file record-by-record.

##### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ITERR	DF3	Terrain record.
NX	DF3	Number of cells in the X direction.
NY	DF3	Number of cells in the Y direction.
NTERR	DF3	Size of terrain cell.

c. Output Variables. Refer to input variables.

6. ROUTINE WETHLD:

a. Purpose. WETHLD creates the weather data file, data file 4, and loads the file with the weather data read from punched cards.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IDUMP	ONE	Flag indicating that only a dump is desired.
IDAY	Card	Total number of days to be loaded.
ISEC	Card	Total number of weather sectors to be loaded.
ISTOP	Card	End-of-deck indicator; 999 indicates end of data.
IAR(9)	Card	Weather record (nine words) in the following sequence: visibility index, cloud cover, temperature, precipitation index, temperature gradient, relative humidity, wind velocity, wind direction, and fog index.
MXWETH(9)	DF04	Weather record (see IAR).

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
IAR(9)	DF04	Weather record (see IAR, input variable).
MSEC	Print	Weather sector number.
MDAY	Print	Day.
MHRFRM	Print	End of last hour.
MHRTO	Print	End of this hour.
MXWETH(9)	Print	Weather record (see IAR, input variable).
IREF	Print	File record number.

d. Logical Flow (Figure II-5-B-4):

(1) Block 1. The indicator IDUMP is located in common ONE and must be previously set by another routine to the value DUMP if a dump without loading is desired. If only a dump is desired, transfer control to block L700.

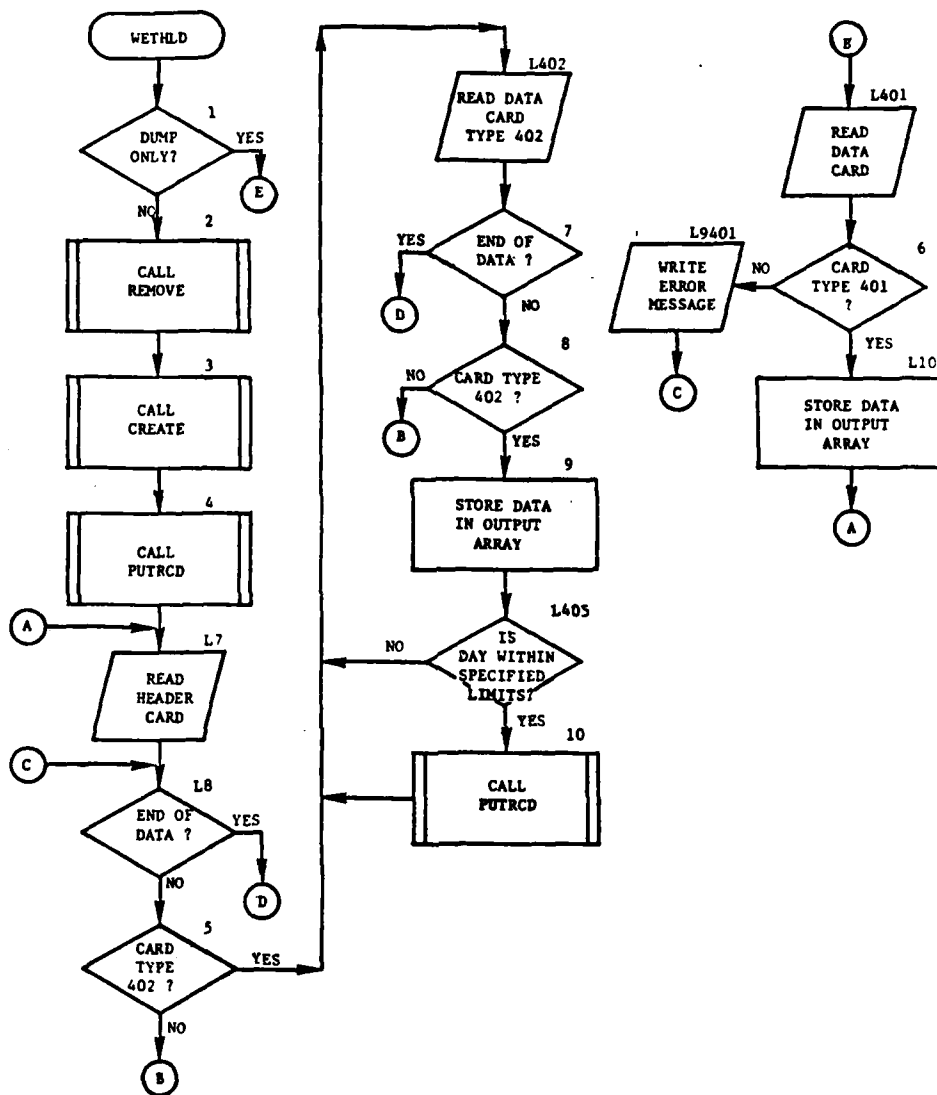


Figure II-5-B-4. Routine WETHLD (Continued on Next Page)

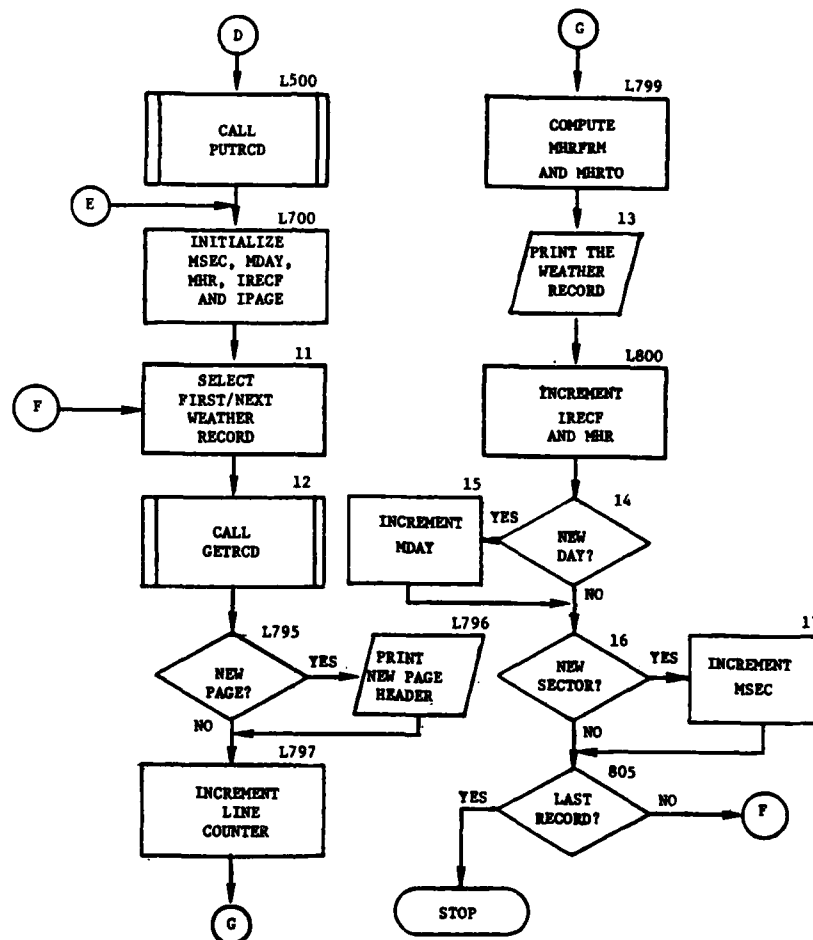


Figure II-5-B-4. Routine WETHLD (Concluded)

(2) Blocks 2, 3, and 4. The previous data file 4 is removed, if one exists, and a new data file 4 is created and zeroed.

(3) Block L7. A header card is read that indicates the card type to be read (401 or 402).

(4) Blocks L8 and 5. The card is checked for end-of-file indicator (9999 in columns 73-76) and then for card type 402. If an end of data was encountered, transfer control to block L500. If a card type 402 was encountered, transfer control to block L402.

(5) Blocks L401 and 6. A data card is read and checked for card type. If the card type is 401, transfer control to block L10.

(6) Block L9401. If the card type just read is not 401, an error message is written, and control returns to block L8.

(7) Block L10. The data from card type 401 are stored in the output array and control returns to block L7.

(8) Block L402. When a type 402 header card is read, control passes to block L402 and another data card is read.

(9) Blocks 7 and 8. The card is checked for end-of-file and for card type 402. If an end of data is encountered, transfer control to block L500. If card type 402 is not encountered, transfer control to block L401.

(10) Block 9. If the card type was 402, the data are stored in the output array.

(11) Blocks L405 and 10. If the day number is within the specified range, put the weather data on data file 4; then transfer control to block L402 to read the next card.

(12) Block L500. Put weather zone coordinates, sunrise and sunset times, and total number of days on data file 4.

(13) Block L700. The sector number (MSEC), day number (MDAY), hour number (MHR), and record number (IRECF) are initialized to one. The line counter (IPAGE) is set to zero.

(14) Block 11. This block is the top of the loop to bring in each weather record from data file 4 for printing.

(15) Block 12. The record at record number IRECF is brought into MXWETH from data file 4.

(16) Blocks L795 and L796. A new page header providing the title and column headings is printed if the line counter (IPAGE) equals zero.

(17) Block L797. The line counter (IPAGE) is incremented by one.

(18) Block L799. The hour numbers MHRFROM and MHRT0 are computed from the cumulative hour number MHR.

(19) Block 13. The line for the weather recor. is printed from the following output variables: MSEC, MDAY, MHRFRM, MHRT , MXWETH(1-9), and IRECF.

(20) Block L800. IRECF and MHR are incremented to go to the next record which corresponds to the next hour.

(21) Blocks 14 and 15. If MHR exceeds 24, MHR is reset to one, and MDAY is incremented by one.

(22) Blocks 16 and 17. If MDAY exceeds 14, MDAY is reset to one, and MSEC is incremented by one.

(23) Block 805. If the last weather record has not been printed, transfer control to block 11; otherwise, control returns to the calling routine.

## APPENDIX C

### ENVIRONMENTAL DATA LOAD OUTPUT DESCRIPTIONS

The input loaded into the weather file produces a formatted printout as shown in Figure II-5-C-1. Each weather zone may contain a maximum of 14 days of weather data; therefore, regardless of the number of days of weather input for each zone, 14 pages of printout, i.e., one for each day; will be printed.

[illegible]

Figure II-5-C-1. Weather File Printout



APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR ENVIRONMENTAL REPRESENTATION

(AVAILABLE UNDER SEPARATE COVER)

NOTES

II-5-D-2

## CHAPTER 6

### UNIT GEOMETRY DATA INPUT PROGRAMS

1. INTRODUCTION. The Unit Geometry data input programs create and load the constant data files describing unit dimensions and distributions for each type of unit performing seven activities. The data specify the width and depth of the unit, the number of bands in which the personnel and equipment are distributed, and the fractional distribution of personnel and equipment among the bands for each activity.

#### 2. ROUTINES:

a. General. The Unit Geometry data input programs consist of the three routines described below.

b. Routine DIMLD. The controlling executive function is performed by routine DIMLD. This routine calls routine LOAD28 and routine DUMP28 in that order.

c. Routine LOAD28. The unit dimension and distribution data file is created and loaded by routine LOAD28. This routine also reads and edits the data cards.

d. Routine DUMP28. Routine DUMP28 produces a formatted dump of the contents of the unit dimension and distribution data file.

3. FILES. The following data file is loaded by these routines.

. Unit dimension and distribution - data file 28

NOTES

II-6-2

## APPENDIX A

### UNIT GEOMETRY DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for unit geometry are documented in Appendix A, Battlefield and Unit Geometry Input Requirements, to Chapter 5 of the Period Processor (Section IV).

NOTES

II-6-A-2

## APPENDIX B

### UNIT GEOMETRY DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. Unit geometry data are loaded onto data file 28 by routines described in this appendix. Routine DIMLD is the controlling routine, LOAD28 reads the data cards and loads data file 28, and DUMP28 produces formatted tables displaying the data. Unit dimensions, number of bands, and distribution of personnel and equipment among the bands are principal data elements.

#### 2. ROUTINE DIMLD:

a. Purpose. DIMLD controls loading of data file 28 with unit dimensions and item distribution data from cards. DIMLD calls LOAD28 to load the data and DUMP28 to print the data.

b. Input Variables: None.

c. Output Variables: None.

d. Processing Description. DIMLD performs three significant operations in sequence: calls LOAD28 to load the data, calls DUMP28 to print the data, and prints the file name table.

#### 3. ROUTINE LOAD28:

a. Purpose. LOAD28 loads the unit dimensions and item distribution data onto data file 28 from cards.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
KARD(20)	Card	Data card image buffer.
IFILE	Card	File number (28).
IFORM	Card	Card format number (01-04).
ISIDE	Card	Force Designator (B or R).
IUSN	Card	Unit type designator (UTD) group assignment number (1-189).
UTDSET(12)	Card	List of one to twelve UTDs assigned to a UTD group.
NEOH	Card	Number of items with nonuniform distributions (1-20), number of entries in EOHLST.
EOHLST(20)	Card	List of item codes of items with nonuniform distribution.
UWIDTH(7)	Card	Unit width for seven activity indexes.

<u>Name</u>	<u>Source</u>	<u>Contents</u>
UDEPTH(7)	Card	Unit depth for seven activity indexes.
NBANDS(7)	Card	Number of bands for seven activity indexes.
K	Card	Scale factor for multiplying times UWIDTH and UDEPTH before loading on data file 28.
JECH	Card	Leftmost character of the three-character equipment item code; used to recognize the entry "P" as indicating personnel.
IECH	Card	Two rightmost characters of the three-character item code.
IACH(4)	Card	List of one to four activity indexes.
PCENTS(4,4)	Card	The percentage of the total amount on hand of the item for each of the four possible bands and the four activity indexes listed in IACH.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
KARD(20)	Print	Data card image buffer.
UDDTBL(189)	DF28	Unit dimensions and item distribution tables for one UTD group, one data file 28 record.
NECH		UDDTBL(1) Number of entries in ECHLST.
ECHLST(20)		UDDTBL(2). List of item codes of equipment items with nonuniform distributions.
UWIDTH(7)		UDDTBL(22). Unit width for seven activity indexes.
UDEPTH(7)		UDDTBL(29). Unit depth for seven activities.
NBANDS(7)		UDDTBL(36). Number of bands for seven activity indexes.
PERSD(7)		UDDTBL(43). Packed distribution words for seven activity indexes.
EOHD(7,20)		UDDTBL(50). Packed distribution words for seven activity indexes and up to 20 equipment items as listed in ECHLST.



d. Logical Flow (Figure II-6-B-1):

(1) Blocks L50, 1, 2, 3, and L63. The logic in these blocks reads each data card using a character format, assigns a sequence number to each card image, prints a card image for each card, and stores each card image on a temporary storage file for subsequent processing and loading onto data file 28.

(2) Block 4. Rewind the temporary storage file to begin load program logic.

(3) Blocks L100, 5, 6, and 7. The outdated data file 28 is removed, a new data file 28 is created, and directory and record buffer arrays are zeroed and initialized before loading the data.

(4) Block L120. This block is the beginning of the loop to read each card image and store the data on data file 28.

(5) Block L200. If the card file number is not equal to 28 control transfers to block 8.

(6) Block L205. The record buffer array, UDDTBL, is zeroed prior to entering data.

(7) Block L210. Logic of this block branches control to the correct set of logic to read the data with the proper format and to interpret and store it accordingly. The format code is contained in columns 74 and 75 of each data card and is part of the card identification.

(8) Block L999. If an error is detected in the data card formats or in card sequencing, control branches to this block, an error message is produced, and the load program is terminated.

(9) Block 8. The file number is checked for an end of file entry of 99 which indicates normal termination of the load procedure. If the entry is not equal to 99, an error message is produced indicating an improper card file number, and control transfers to block L999.

(10) Block L350. Following normal load procedure, when the last data card image has been entered into data file 28, the file directory records are entered on data file 28.

(11) Block L120. Logic for loading data from card identification 2801 begins by reading the card image from the temporary storage file with a 2801 format. This is the unit dimension and distribution grouping data.

(12) Blocks 9, 14, 17, and 19. Logic in these blocks ensures that the format used to read the data corresponds to the format code on the data card. If not, control is transferred to block L200.

(13) Block 10. The force indicator data entry is checked to ensure that the entry is either R or B. If not, control is transferred to block L999.

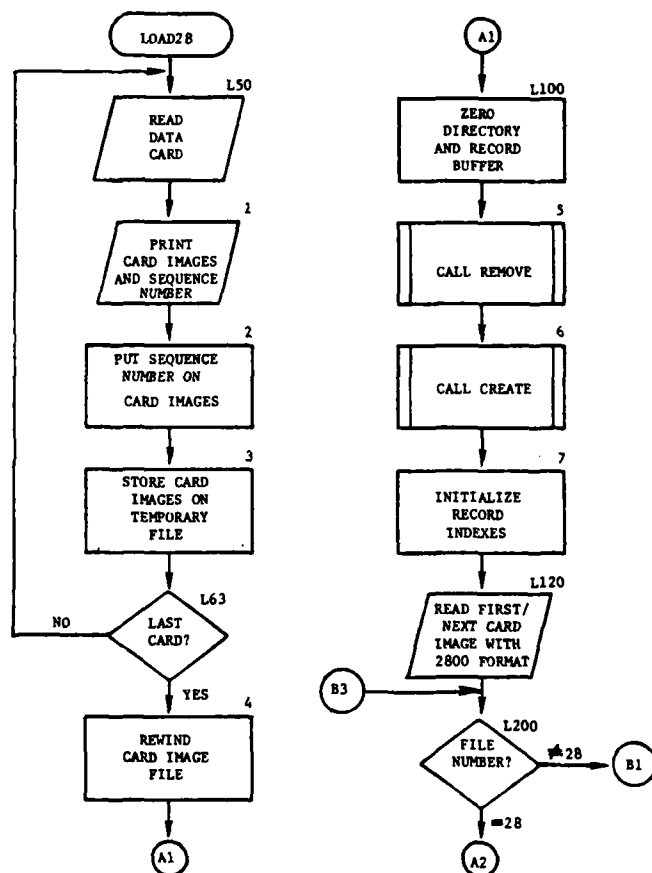


Figure II-6-B-1. Routine LOAD28 (Continued on Next Page)

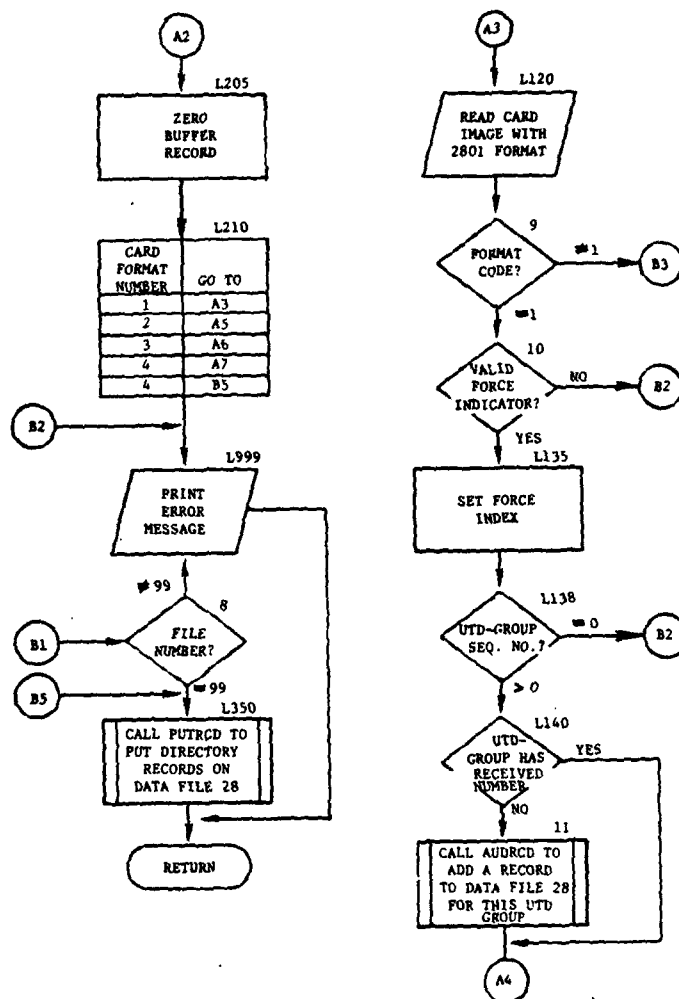


Figure II-6-B-1. Routine LOAD28 (Continued)

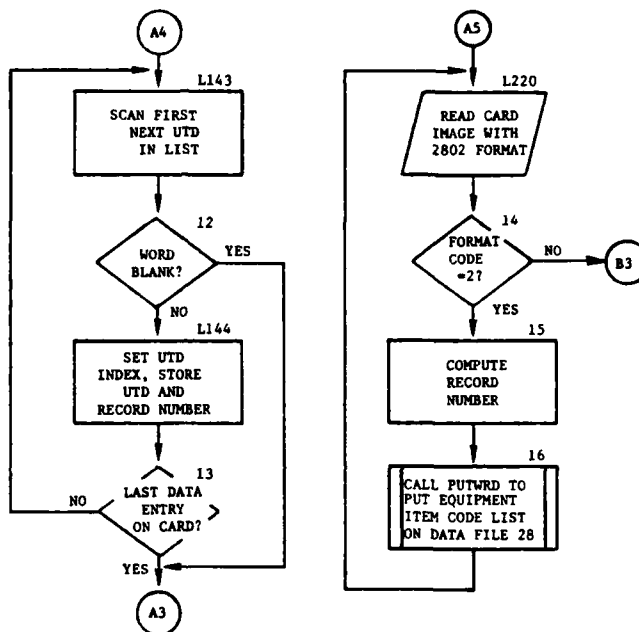


Figure II-6-B-1. Routine LOAD28 (Continued)

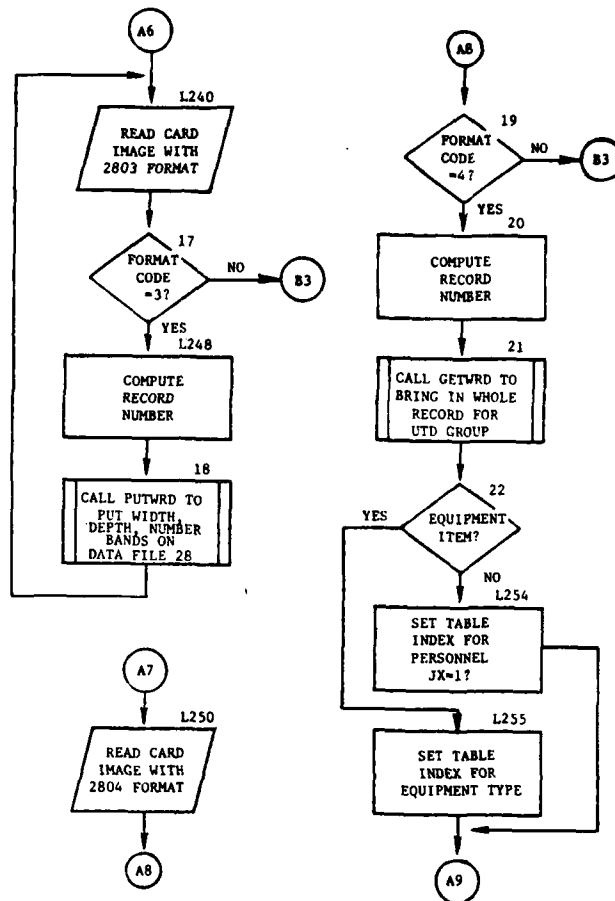


Figure II-6-B-1. Routine LOAD28 (Continued)

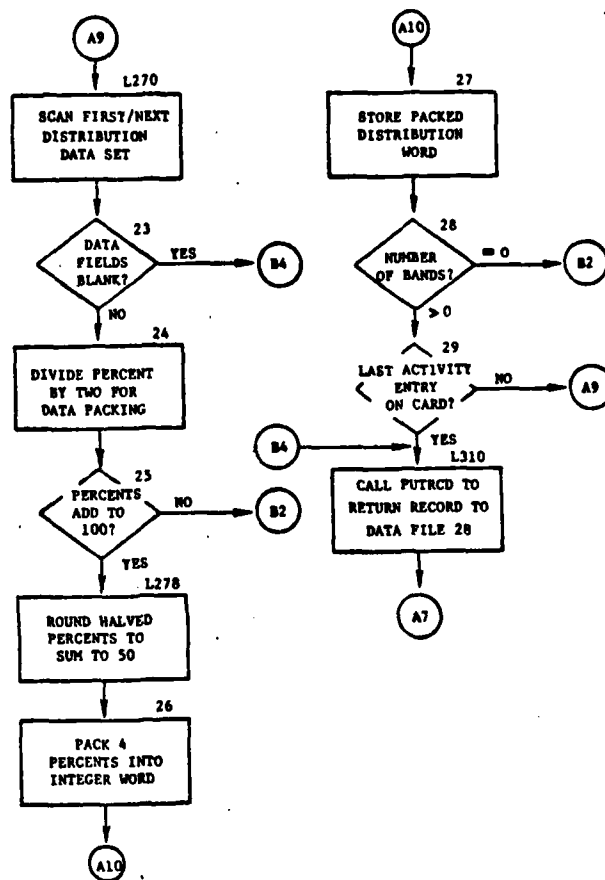


Figure II-6-B-1. Routine LOAD28 (Concluded)

(14) Block L135. The force indicator index, IFORCE, is set to 1 for Blue and to 2 for Red.

(15) Block L138. Each data card must contain the UTD group sequence number; otherwise, control is transferred to block L999.

(16) Blocks L140 and 11. As successive UTD groups identified on the data cards are loaded, each is assigned a record location on data file 28. The record index is stored in the directory array, IRECN.

(17) Blocks L143, 12, L144, and 13. Logic contained in these blocks reads each UTD in the data list, increments a UTD counter index, and stores the UTD with the record pointer index in the directory array, DRCTRY. The maximum number of UTDs allowed per force is 189.

(18) Block L220. Logic beginning with this block reads data entries on card identification 2802. Data contained therein are the list of equipment items that are to be nonuniformly distributed in the bands of units which have UTDs belonging to this UTD group. The data are stored in EOHLST(20) located at UDDTBL(2).

(19) Blocks 15 and 16. The record number on data file 28 is obtained by using the force index and UTD group sequence number as indexes to the array IRECN that was established previously. The data allow a maximum of 21 entries, the first of which is the number of items nonuniformly distributed, NEOH. Subsequent entries are the item codes of the items that are distributed, EOHLST. These entries are loaded onto data file 28.

(20) Block L240. Logic beginning with this block loads band and unit dimension data, NBANDS(7), UWIDTH(7), and UDEPTH(7), onto data file 28. The number of bands, NBANDS; width of unit, UWIDTH; and depth of unit, UDEPTH, are entered for each of seven activities. A scale factor, K, if entered in card columns 70-72, is applied to every width and depth entry for the UTD group sequence number.

(21) Blocks L248 and 18. The record index is obtained from IRECN, and the data arrays are loaded onto data file 28. Control then transfers to block L240.

(22) Block L250. This block begins the loop for loading the distribution data for personnel and equipment by bands. The data array filled is EOHD(7,20) and contains the packed distribution words for each of the equipment items in up to seven activities. The array is located at UDDTBL(50). Control is then transferred to block 19.

(23) Blocks 20 and 21. Once the record index has been established, the entire record is brought into the array, UDDTBL, and the distribution data are added to the array by subsequent logic.

(24) Blocks 22, L254, and L255. The first column of the equipment item code is read under a character format and the second two columns under an integer format. If the distribution is for personnel, the equipment item code entry has a character "P" in the first column or a zero in the third column. Data entries are loaded into UDDTBL(7,JX) where JX equals one for personnel and JX equals two through 20 for equipment items.

(25) Blocks L270 and 23. Each item is scanned beginning with this block until a blank entry for an activity code indicates the end of the data for this UTD group; then, control is transferred to block L310.

(26) Blocks 24, 25, L278, 26 and 27. The logic contained in these blocks performs two functions. The first is to check the data entries to ensure that the percentages sum to 100; if they do not, control transfers to block L999. The second is to pack the integer distribution percents into a single word for storage on data file 28. To store as a single word, each percent is first divided by two and rounded such that when unpacked, the sum will be 100. The percentages are each packed into six bits of a word. The packed record is stored in PCHAR(7) for each activity.

(27) Block 28. If the number of bands data entry is not greater than zero, control transfers to block L999.

(28) Block 29. If the last activity on the card has not been processed control returns to block L270.

(29) Block L310. After the last data entry on the card has been processed, the entire record array, UDDTBL, is replaced on data file 28 and control transfers to block L250.

#### 4. ROUTINE DUMP28:

a. Purpose. DUMP28 produces a printed output listing of the contents of data file 28, unit dimensions and personnel and equipment distribution data.

##### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
UDDTBL(189)	DF28	Unit dimension and distribution table.
NEOH		UDDTBL(1). Number of items with nonuniform distributions.
EOHLST(20)		UDDTBL(2). List of item codes of items with nonuniform distributions.
UWIDTH(7)		UDDTBL(22). Unit width for seven activity indexes.



<u>Name</u>	<u>Source</u>	<u>Contents</u>
UDEPTH(7)		UDDTBL(29). Unit depth for seven activity indexes.
NBANDS(7)		UDDTBL(36) Number of bands for seven activity indexes.
PERSD(7)		UDDTBL(43). Personnel distribution densities, packed, for seven activities.
EOH(7,20)		UDDTBL(50). Equipment item distribution densities, packed, for seven activities.
DTABL(756)		Data file 28 directory table.
DRCTRY(189,4)		DTABL(1). Data file 28 directory table.
UTDSB(189)		DTABL(1). Blue force UTD array.
UTDSR(189)		DTABL(379). Red force UTD array.
RECNB(189)		DTABL(190). Record pointer index for Blue data records corresponding to UTDs stored at same position in UTDSB.
RECNR(189)		DTABL(568). Record pointer index for Red data records corresponding to UTDs stored at same position in UTDSR.

c. Output Variables. Same as input variables.

d. Logical Flow (Figure II-9-B-2):

(1) Block 1. The data file 28 directory table, stored in the first four records on the file, is brought in and stored in DTABL.

(2) Block 2. The directory data stored in DTABL are printed for both forces.

(3) Block 3. This block initiates the loop on the two forces. The Blue force data are printed first, followed by the Red force data.

(4) Block L200. This block begins the loop to obtain data records from the file. Each record contains the data for one UTD group within the specified force being dumped.

(5) Block L400. Initial printout for each data record contains the record number, force identification, and list of UTDs characterized by the data in this UTD group record.

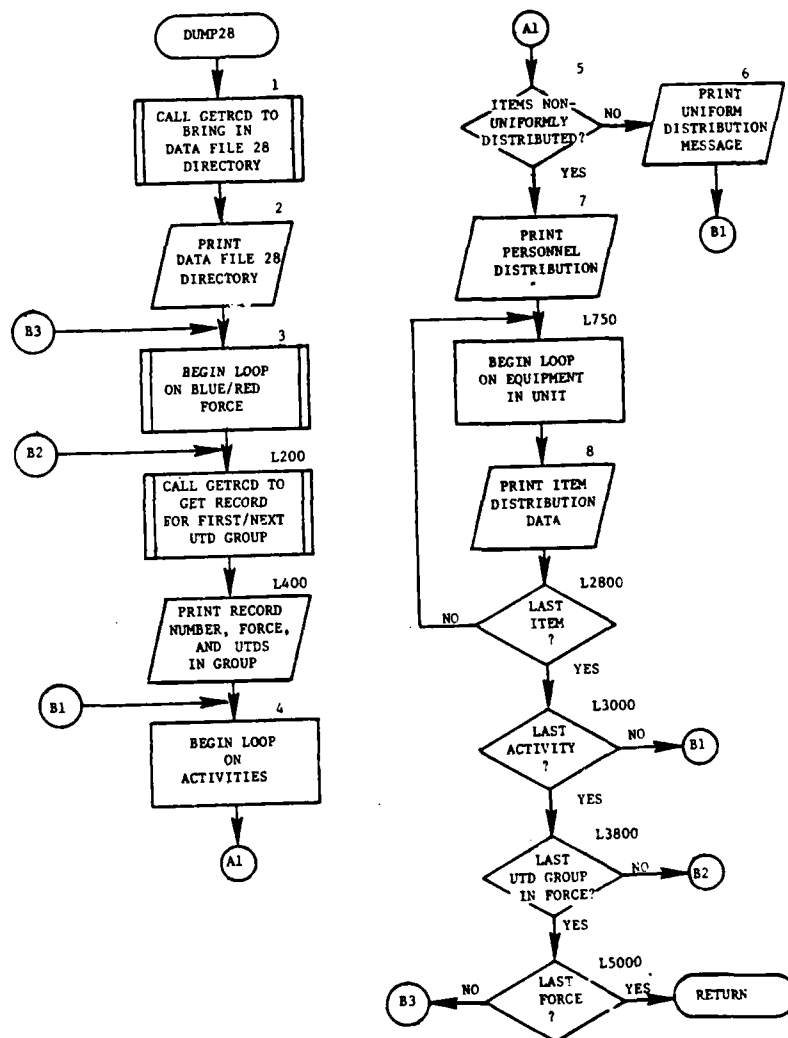


Figure II-6-B-2. Routine DUMP28

AD-A112 658

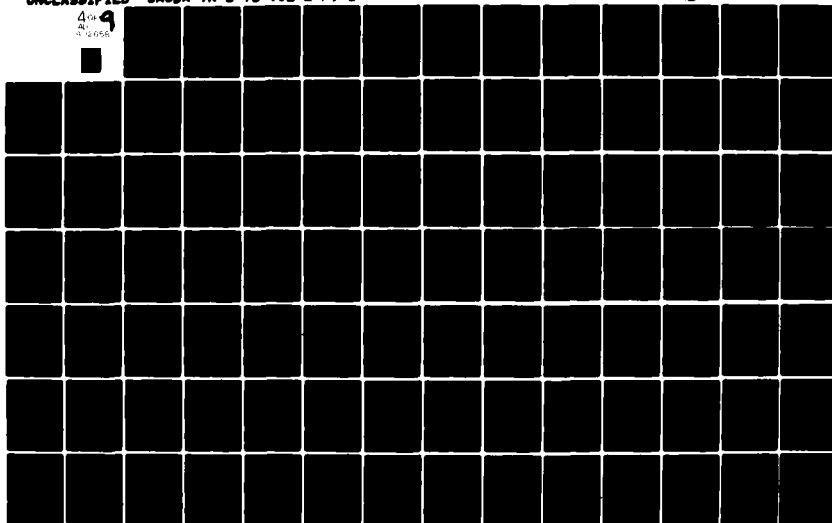
ARMY COMBINED ARMS COMBAT DEVELOPMENTS ACTIVITY FORT--ETC F/8 18/7  
DIVVAG MODEL DOCUMENTATION. VOLUME II. PROGRAMMER/ANALYST MANUAL--ETC(U)  
JUL 76

UNCLASSIFIED

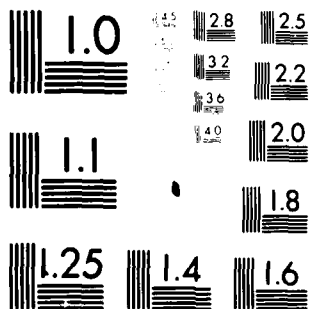
CACDA-TR-8-76-VOL-2-PT-1

NL

4-9  
3-12658



11265



MICROCOPY RESOLUTION TEST CHART  
NBS 1963-A

(6) Block 4. The data in UDDTLB for the UTD group are printed for each activity. This block is the beginning of the loop on the activity index of the dimension and distribution data.

(7) Blocks 5, 6, and 7. Logic contained in these blocks screens the data for each activity and determines if nonuniform distributions for personnel or equipment items exist. If they do not, uniform distribution messages are printed; otherwise, the distribution data for personnel are printed and item distributions are computed in subsequent blocks. Personnel data are in array PERSD(7).

(8) Block L750. Scanning of equipment items nonuniformly distributed begins with this block.

(9) Block 8. The distribution data, including equipment item code and percent of equipment in each band, are printed for the equipment item being considered. Data are taken from array EOHD(7,20).

(10) Block L2800. This block completes the loop on equipment index in EOHD array.

(11) Block L3000. This block completes the loop on activity index in EOHD and PERDS arrays.

(12) Block L3800. This block completes the loop on UTD group (also data record) for the specified force.

(13) Block L5000. This block completes the loop on Blue and Red forces.

NOTES

II-6-B-14

## APPENDIX C

### UNIT GEOMETRY DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. Three different printouts of constant data inputs are generated after the data have been read into the files. These printouts are as follows:

<u>Printout Title</u>	<u>Data Source</u>
80-Column Card Image Data File 28	Card data deck
Unit Type Designator Record Number Directory	Data file 28
Unit Descriptions	Data file 28

2. EIGHTY-COLUMN CARD IMAGE, DATA FILE 28. The format is illustrated in Figure II-6-C-1. At the far left, top line, of this figure are the characters B001, indicating that the data are for Blue forces and 001 is the index number assigned to that grouping of UTDs. In this case there is only one UTD in the group, PFL as shown at the right of the B001. At the far right is card 1, indicating that the sequence number of the card is 1, and therefore is the first card in this identifier. The identifier of this card is in the second column from the right. The figures 2801 indicate that these are data for data file 28 in the 01 format.

3. UNIT TYPE DESIGNATOR RECORD NUMBER DIRECTORY. The second printout in the series lists the UTDs and the record number that contains the data for each UTD. This printout is shown in Figure II-6-C-2.

4. UNIT DESCRIPTIONS (Figure II-6-C-3). The data in this format are for only one force (in this case, the Blue force), and contain the description of the unit in terms of the dimensions and distribution of personnel and equipment as representative of stay, move, fire, and attack activities.

FILE-23 DATA CARD IMAGES

9001 PELL	2801 CARD	1
9002 JELL	2801 CARD	2
9003 ITCT	2801 CARD	3
9004 KHL	2801 CARD	4
9005 JTL	2801 CARD	5
9006 KFL	2801 CARD	6
9007 KDL	2801 CARD	7
9008 JLL	2801 CARD	8
9009 KHL	2801 CARD	9
9010 KHL	2801 CARD	10
9011 KHL	2801 CARD	11
9012 KHL	2801 CARD	12
9013 KHL	2801 CARD	13
9014 KHL	2801 CARD	14
9015 KHL	2801 CARD	15
9016 KHL	2801 CARD	16
9017 KHL	2801 CARD	17
9018 KHL	2801 CARD	18
9019 KHL	2801 CARD	19
9020 KHL	2801 CARD	20
9021 KHL	2801 CARD	21
9022 KHL	2801 CARD	22
9023 KHL	2801 CARD	23
9024 KHL	2801 CARD	24
9025 KHL	2801 CARD	25
9026 KHL	2801 CARD	26
9027 KHL	2801 CARD	27
9028 KHL	2801 CARD	28
9029 KHL	2801 CARD	29
9030 KHL	2801 CARD	30
9031 KHL	2801 CARD	31
9032 KHL	2801 CARD	32
9033 KHL	2801 CARD	33

II-6-C-2

Figure II-6-C-1. Data File 28 80-Column Card Image



RECORDS 1 THRU 4 CONTAIN THE FILE-29 DIIRECTORY FOR THE RECORD NUMBER OF EACH UTO GROUP

[illegible]

**Figure II-6-C-2. Unit Type Designator Record Number Directory**

RECORD NUMBER 37 CONTAINS THE DESCRIPTION OF BLUE UNITS WITH THESE UTD'S :

ICCT

ACTIVITY INDEX 1 (STAY ) WIDTH = 200 DEPTH = 200 NO. BANDS = 1

PERSONNEL DISTRIBUTED UNIFORMLY AMONG BANDS  
EQUIPMENT DISTRIBUTED UNIFORMLY AMONG BANDS

ACTIVITY INDEX 2 (MOVE ) WIDTH = 25 DEPTH = 1800 NO. BANDS = 1

PERSONNEL DISTRIBUTED UNIFORMLY AMONG BANDS  
EQUIPMENT DISTRIBUTED UNIFORMLY AMONG BANDS

ACTIVITY INDEX 3 (FIRE ) WIDTH = 0 DEPTH = 0 NO. BANDS = 0

PERSONNEL DISTRIBUTED UNIFORMLY AMONG BANDS  
EQUIPMENT DISTRIBUTED UNIFORMLY AMONG BANDS

ACTIVITY INDEX 4 (ATTACK ) WIDTH = 200 DEPTH = 400 NO. BANDS = 2

PERSONNEL	DISTRIBUTION :	31310000 :	50 %	50 %
ITEM CODE 8	DISTRIBUTION :	620000 :	0 %	100 %
ITEM CODE 58	DISTRIBUTION :	620000 :	0 %	100 %
ITEM CODE 63	DISTRIBUTION :	62000000 :	100 %	0 %
ITEM CODE 65	DISTRIBUTION :	31310000 :	50 %	50 %
ITEM CODE 66	DISTRIBUTION :	125000000 :	20 %	80 %
ITEM CODE 67	DISTRIBUTION :	620000 :	0 %	100 %
ITEM CODE 75	DISTRIBUTION :	620000 :	0 %	100 %

ALL OTHER ITEMS DISTRIBUTED UNIFORMLY AMONG BANDS

Figure II-6-C-3. Unit Descriptions

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR UNIT GEOMETRY

(AVAILABLE UNDER SEPARATE COVER)

II-6-D-1

NOTES

II-6-D-2

## CHAPTER 7

### INTELLIGENCE AND CONTROL DATA INPUT PROGRAMS

1. INTRODUCTION. The Intelligence and Control data input programs create and load the constant data files describing sensor characteristics and decision and communication tables. In addition, they create and initialize the dynamic unit intelligence files.

2. ROUTINES:

a. General. The Intelligence and Control data input programs consist of the four routines described below.

b. Routine INCSLD. Routine INCSLD reads the appropriate data from input cards and loads the decision and communication data tables. This routine also creates and initializes the dynamic unit intelligence files and the redundant report file. This routine calls routine SET20 to load the sensor data and routine DUMP36 to produce the formatted dump of the decision and communication tables.

c. Routine DUMP36. The decision and communication data are displayed in a formatted dump by routine DUMP36.

d. Routine SET20. The characteristics of all sensors to be included in the DIVWAG system are read from data cards and stored on the sensor data file by routine SET20. Routine DUMP20 is called at the completion of this process.

e. Routine DUMP20. A complete formatted dump of the contents of the sensor data file is produced by routine DUMP20.

3. FILES. The following data files are created and loaded by these routines:

. Sensor characteristics	-	data file 20
. Decision and communication tables	-	data file 36
. Unit intelligence files	-	data files 41, 42, and 43
. Redundant report file	-	data file 44

NOTES

II-7-2

## APPENDIX A

### INTELLIGENCE AND CONTROL DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for intelligence and control are documented in Appendix A, Intelligence and Control Model Input Requirements, to Chapter 6 of the Period Processor (Section IV).

NOTES

II-7-A-2



## APPENDIX B

### INTELLIGENCE AND CONTROL DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. Routines INCSLD and SET20 are described in this appendix. Associated dump routines, DUMP36 and DUMP20, provide a formatted listing of data files 36 and 20 respectively.

2. ROUTINE INCSLD:

a. Purpose. INCSLD loads data file 36 with Intelligence and Control Model data and creates and zeroes data files 41, 42, 43, 44, and 45. The data loaded by this routine are the decision tables, delay times, unit size estimate and radius tables, and equipment type codes.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
C3601	Card	Card type 3601 containing delay times versus unit type designators (UTD), used to load UTDT (UTD table), PDT (processing delay time), and DDT (decision delay time).
C3602	Card	Card type 3602 consisting of equipment type code classification data.
C3603	Card	Card type 3603 containing data to load USET (unit size estimate table) and URT (unit radius table).
C3604	Card	Card type 3604 consisting of the threshold matrices for the dissemination of intelligence and associated information flow delay times.
ISA	DF41 DF42 DF43 DF44 DF45	An array filled with zeroes utilized to zero data files 41, 42, 43, 44, and 45.
FILE36	DF36	Data file 36 data loaded from cards by INCSLD routine.

d. Logical Flow (Figure II-7-B-1):

(1) Block 1. GETFLE is called to obtain the file name table (IFNT) to utilize the DIVWAG input/output package.

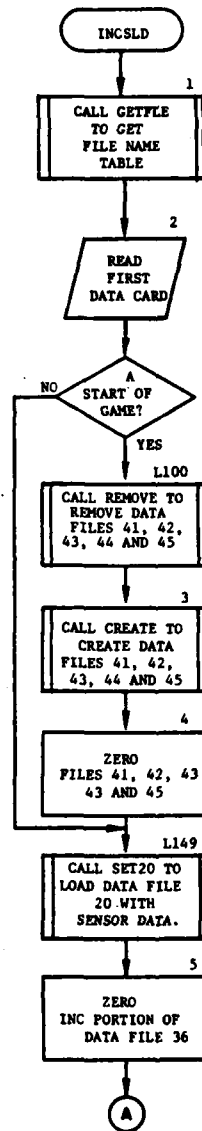


Figure II-7-B-1. Routine INCSLD (Continued on Next Page)

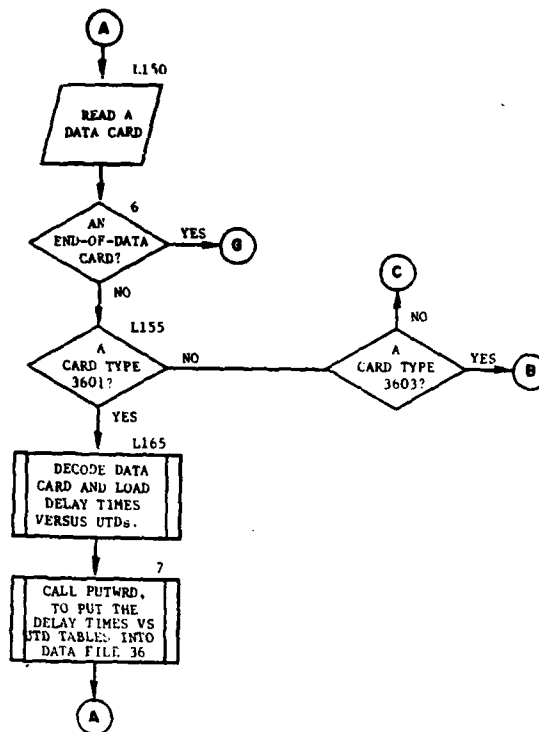


Figure II-7-B-1. Routine INCSLD (Continued)

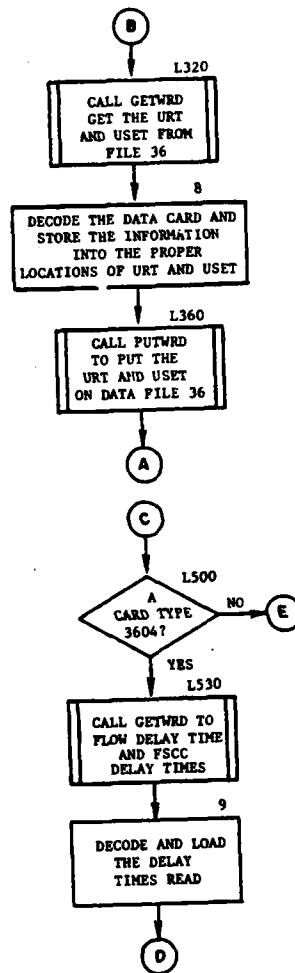


Figure II-7-B-1. Routine INCSLD (Continued)

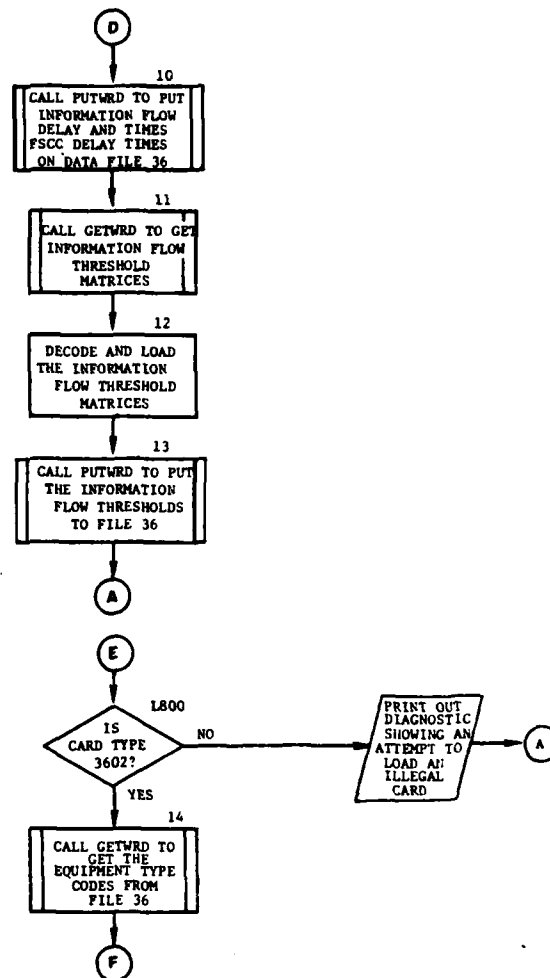


Figure II-7-B-1. Routine INCSLD (Continued)



Figure II-7-B-1. Routine INCSLD (Continued)

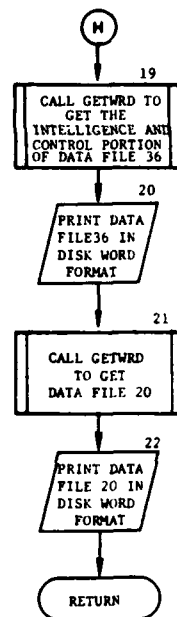


Figure II-7-B-1. Routine INCSLD (Concluded)

(2) Block 2. Read the first data card to determine if it is a game or period card. If it is a period card, control goes to block L149.

(3) Block L100. Routine REMOVE, is called to remove data files 41, 42, 43, 44, and 45 from the DIVWAG data system.

(4) Block 3. These same files are allocated the proper number of records and record size through a call to CREATE.

(5) Block 4. Data files 41, 42, 43, 44, and 45 are zeroed for proper utilization within the Intelligence and Control Model.

(6) Block L149. Through a call to the routine SET20 the sensor constant data are loaded into data file 20.

(7) Block 5. Zero the Intelligence and Control portion of data file 36 appropriately for a game or period load. (All data file 36 word locations are different for game or period loads.)

(8) Block L150. Read a data card under a (20A4) format.

(9) Block 6. If this is an end-of-data card (9999), control goes to block L980.

(10) Block L155. If this is not a card type 3601, control then goes to block L300.

(11) Block L165. Decode the data card into proper format to load the independent variable unit type designator table, IUTD, the decision delay time table, IDDT, and the processing delay time table, IPDT.

(12) Block 7. PUTWRD is called to place the delay times versus UTD arrays [IUTD(50), IDDT(50), IPDT(50)] to data file 36 and control returns to block L150.

(13) Block L300. If this is not a card type 3603, control goes to block L500.

(14) Block L320. The unit size estimate table USET(7,22), and the unit radius table, URT(7,11) are obtained from data file 36.

(15) Block 8. Decode the data card in the proper format and load the unit size estimate table and the unit radius table with the data from this card.

(16) Block L360. Replace the arrays USET and URT onto data file 36; control returns to block L150.

(17) Block L500. If this is not a card type 3604, control goes to block L800.



(18) Block L530. Obtain from data file 36 the information flow delay times (INFLO(5,5)), the attack helicopter and artillery range limits, and the request delay time for fire support (CAS, DAFS, or artillery).

(19) Block 9. Decode the data card in the correct format and load the delay times and range limits into their proper locations within the arrays.

(20) Block 10. Put the delay times and range limits back on data file 36.

(21) Block 11. Obtain from data file 36 the correct information flow threshold matrix according to the force and echelon in the card.

(22) Block 12. Decode and load into the threshold matrix the threshold values for communicating intelligence to a specific echelon.

(23) Block 13. Put the updated threshold matrix back on data file 36, and return control to block L150.

(24) Blocks L800 and L890. If this is not a card type 3602, print a diagnostic message showing an attempt to load an illegal card type and return control to block L150.

(25) Block 14. Obtain the equipment type code array, IEQPTY(50), from data file 36.

(26) Block 15. Decode and load the equipment type codes according to equipment item codes.

(27) Block 16. Replace the equipment type code array, IEQPTY(50), on data file 36; control returns to block L150.

(28) Block L980. Obtain the unit size estimate table and the unit radius table from data file 36.

(29) Block 17. Adjust these tables to ensure no zeroes are in their columns. (Nonzero tables are required by the Intelligence and Control Model.)

(30) Block 18. Replace the adjusted unit size estimate table and unit radius table on data file 36.

(31) Block L985. Call DUMP36 to print the contents of data file 36 in formatted printout.

(32) Block 19. Obtain the Intelligence and Control Model portion of data file 36.

(33) Block 20. Print the contents of data file 36 in a file word format for programmer use.

(34) Block 21. Obtain the sensor constant data loaded by SET20 from data file 20.

(35) Block 22. Print the contents of data file 20 in a file record format for programmer use.

3. ROUTINE SET20:

a. Purpose. Routine SET20 loads the sensor constant data of data file 20 (i.e., records 201 and above).

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
C2001T1	Card	Card ID 2001, type 1, containing general radar constant data including maximum/minimum ranges, depth of scan, horizontal beam width, vertical beam width, scan rate, minimum radial velocity, distortion factor, detection error, range error, percent of equipment failure, percent of false alarm, power up and down times, and mean time to track.
C2001T2	Card	Card ID 2001, type 2, containing radar degradation factors. The factors are forestation factors, electronic countermeasure factors, and precipitation factors.
C2001T3	Card	Card ID 2001, type 3, consisting of the detection ranges and associated probabilities of detection and recognition of target types. For MTI radars, the following items are the target types: personnel, wheeled vehicles, tanks, armored personnel carriers, tube artillery, artillery missiles, air defense guns, and air defense missiles. For air defense radar, aircraft is the only target type.
C2001T4	Card	Card ID 2001, type 4, used to load constant data for unattended ground sensor fields. These data include target types and associated sensitive radii for three target types: personnel, wheeled vehicles, and tracked vehicles. Also loaded here is the percent of false alarms.

<u>Name</u>	<u>Source</u>	<u>Contents</u>
C2001T5	Card	Card ID 2001, type 5, consisting of the general radar data for countermortar/counterbattery radars which include tracking probability, low angle of fire and associated CEP, high angle of fire and associated CEP, low to high angle cutoff, horizontal beam coverage, vertical beam separators, vertical beam thickness, percent equipment failure, maximum location error, power up and down times, operator performance time, and pick-up point wait time.
C2001T6	Card	Card ID 2001, type 6, containing the detection ranges of the countermortar/counterbattery radar against specific weapon/munition combinations.
C2001T7	Card	Card ID 2001, type 7, utilized to load the range brackets for movement of groundbased sensors.
C2002T1	Card	Card ID 2002, type 1, utilized to load the sensor constant data for air reconnaissance type sensors.
C2002T2	Card	Card ID 2002, type 2, containing the reconnaissance sensor combinations.
C2002T3	Card	Card ID 2002, type 3, utilized to load the light observation helicopter decision table.
C2002T4	Card	Card ID 2002, type 4, containing the item code of the LOH or fixed wing reconnaissance aircraft, personnel on board, and delay time to prepare for subsequent reconnaissance missions.
C2002T5	Card	Card ID 2002, type 5, utilized to load the remaining equipment for a reconnaissance mission unit. Data consist of item codes and associated amounts.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
FILE20	DF20	Records loaded into DIVWAG data file 20 from input cards described above.

d. Logical Flow (Figure II-7-B-2):

- (1) Block 1. Read the first data card. If the load process is data update, control goes to block L99.
- (2) Block 2. Call CREATE to ensure that data file 20 is allocated at the proper size.
- (3) Block 3. Zero the 1000 records of data file 20, and control goes to block L10.
- (4) Block L99. Obtain the 16 records containing the constant data information on UGS fields from data file 20. Control goes to block L10.
- (5) Block L9. Put the four records of constant radar data into data file 20.
- (6) Block L10. Read a data card under (20A4) format.
- (7) Block 4. If this is an end-of-data card (9999), control goes to block L20.
- (8) Block 5. If this is a 2001 card ID, control goes to block L90.
- (9) Block 6. If this is a 2002 card ID, control goes to block L450.
- (10) Block L20. Put the UGS field constant data (16 records) onto data file 20.
- (11) Block 7. Put the sensor directory table onto data file 20.
- (12) Block 8. Call routine DUMP20 to dump data file 20 in readable formats for user, and return control to the calling routine.
- (13) Block L90. Print the card image previously read in format (20A4).
- (14) Block 10. If this is a card type 4 (UGS field constant data), control goes to block L250.
- (15) Block 11. Decode the first parameters on the data card which are used as the keyword to the sensor constant data and determine the proper record number on data file 20.
- (16) Block 12. Get the designated record from data file 20.
- (17) Block L115. If this is not card type 1 (general radar data for MTI or AD radar), control goes to block L200.
- (18) Block L120. Decode the type 1 data card and store the information for general data for MTI or AD radar; control then goes to block L9.

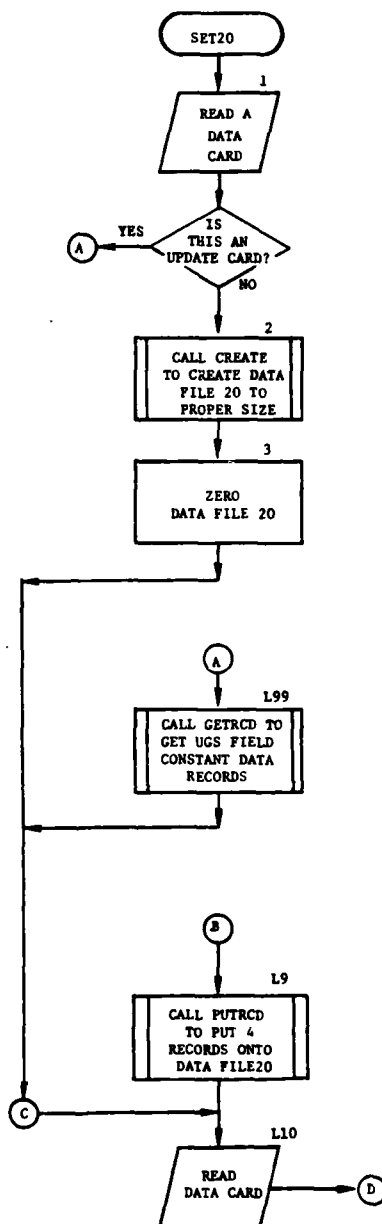


Figure II-7-B-2. Routine SET20 (Continued on Next Page)

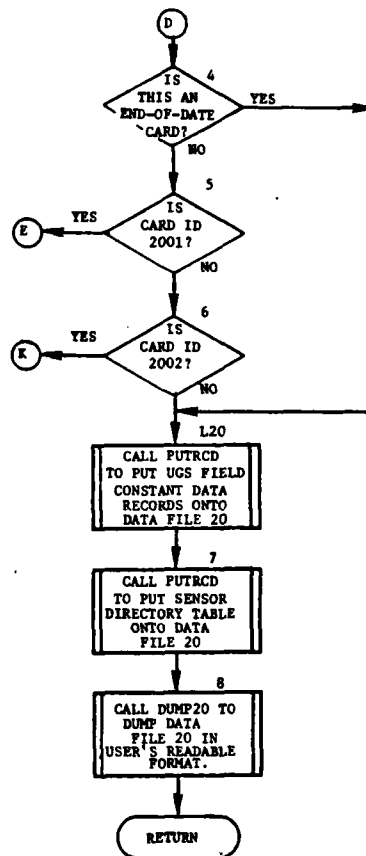


Figure II-7-B-2. Routine SET20 (Continued)

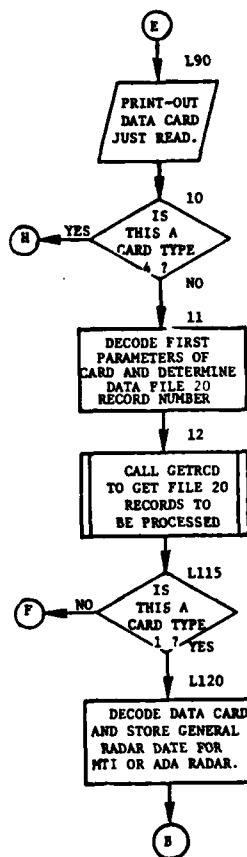


Figure II-7-B-2. Routine SET20 (Continued)

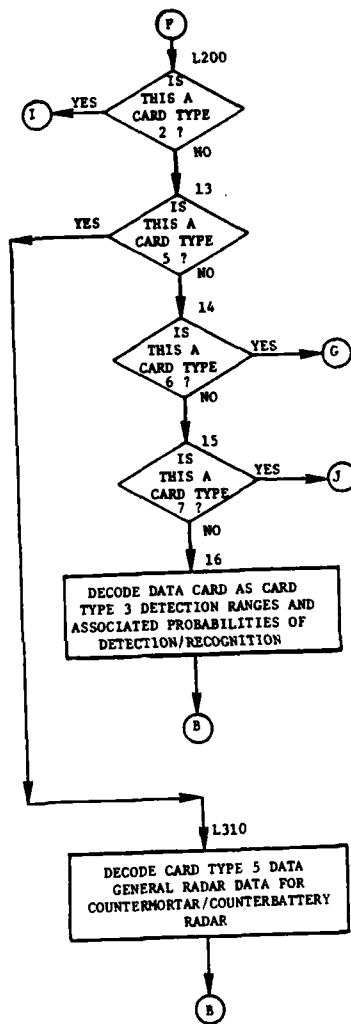


Figure II-7-B-2. Routine SET20 (Continued)



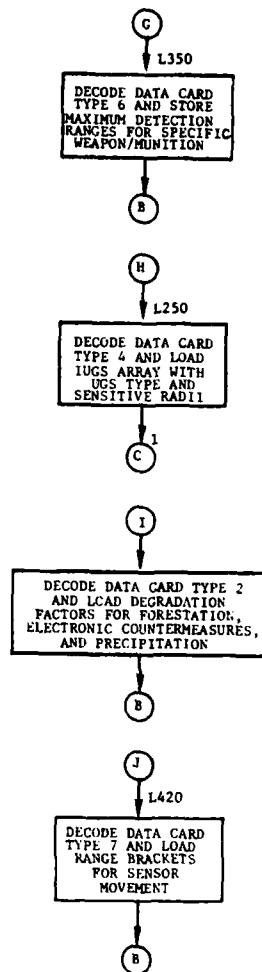


Figure II-7-B-2. Routine SET20 (Continued)

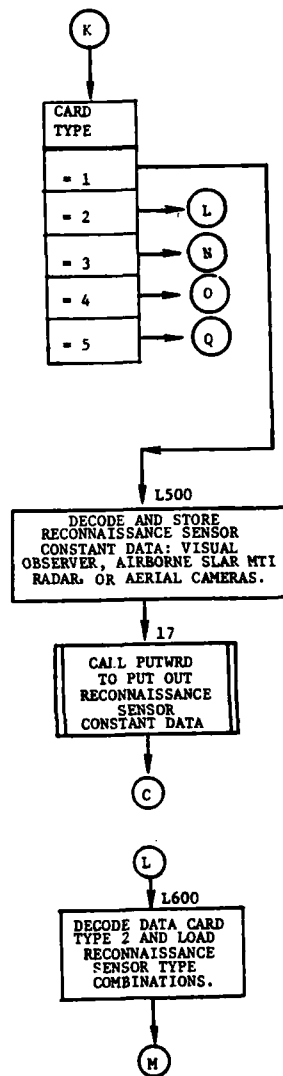


Figure II-7-B-2. Routine SET20 (Continued)

II-7-B-18

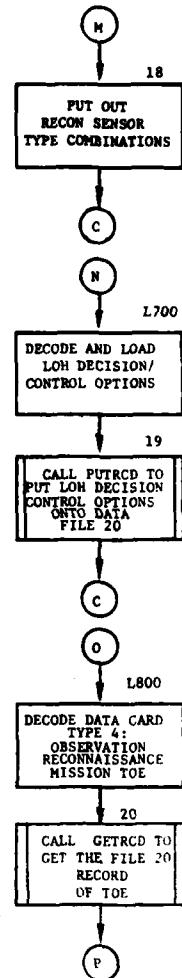


Figure II-7-B-2. Routine SET20 (Continued)

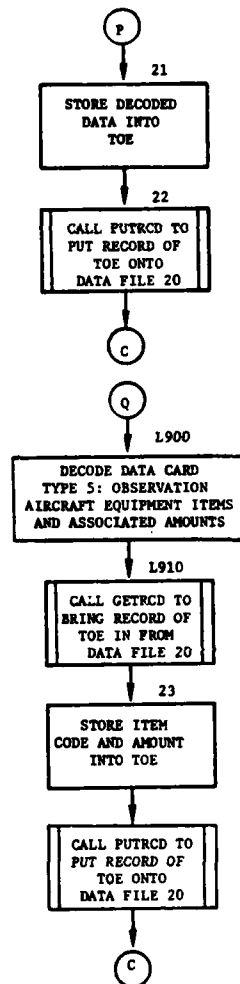


Figure II-7-B-2. Routine SET20 (Concluded)

(19) Block L200. If this is card type 2 (degradation factors), control goes to block L400.

(20) Block 13. If this is card type 5 (general radar data for countermortar/counterbattery radars), control then goes to block L310.

(21) Block 14. If this is card type 6 (maximum detection range for weapon/munition combination), control then goes to block L350.

(22) Block 15. If this is card type 7 (range brackets for sensor movement), control then goes to block L420.

(23) Block 16. Decode data card type 3 (detection ranges and associated probability of detection and recognition) and load these data for storage onto data file 20; control goes to block L9.

(24) Block L310. Decode data card type 5 and set up general countermortar/counterbattery radar data for storage onto data file 20; control goes to block L9.

(25) Block L350. Decode data card type 6 and load maximum detection ranges for specific weapon/munition combinations; control goes to block L9.

(26) Block L250. Decode data card type 4 and load IUGS array with UGS types and associated sensitive radii and percent of false alarms; control goes to block L10.

(27) Block L400. Decode data card type 2 and set up degradation factors for forestation, electronic countermeasures, and precipitation; then, control goes to block L9.

(28) Block L420. Decode data card type 7 and set up the range brackets for sensor movement; and, control goes to block L9.

(29) Block L450. If the card ID 2002 is type 1, control goes to block L500. If the card type is 2, control goes to block L600. If the card type is 3, control goes to block L700. If the card type is 4, control goes to block L800. If the card type is 5, control goes to block L900.

(30) Block L500. Decode and store reconnaissance sensor constant data (e.g., visual observer, airborne SLARMTI radar or aerial cameras).

(31) Block 17. Put the words containing the reconnaissance sensor constant data onto data file 20; control goes to block L10.

(32) Block L600. Decode data card type 2 and set up the reconnaissance sensor type combinations.

(33) Block 18. Put the reconnaissance sensor type combination arrays onto data file 20; and, control goes to block L10.

(34) Block L700. Decode data card type 3 and set up the LOH decision/control option array.

(35) Block 19. Put the LOH decision/control option array onto data file 20; control returns to block L10.

(36) Block L800. Decode data card type 4 and set up aircraft item code, number of personnel aboard, and delay time to prepare for subsequent reconnaissance missions.

(37) Block 20. Get the appropriate mission unit TOE from data file 20.

(38) Block 21. Store aircraft item code, number of personnel, and delay time into TOE array.

(39) Block 22. Put the TOE array back onto data file 20, and control goes to block L10.

(40) Block L900. Decode data card type 5 and set up remaining equipment item codes and associated quantities.

(41) Block L910. Bring the appropriate TOE array from data file 20.

(42) Block 23. Store the item codes and their associated quantities into the TOE array.

(43) Block 24. Put the TOE array back onto data file 20. Control goes to block L10.

## APPENDIX C

### INTELLIGENCE AND CONTROL DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. After the Red and Blue forces constant data input have been entered, a printout of these data is generated. The data in the printouts will be in three classes. The first is an 80-column image of the cards read in the data deck for INC. The second series is the file data, which has been formatted with columnar headings for easy reading. The third set of printouts is the dump of data files 20 and 36. The 80-column card image printout is designed to identify a card when updating is in order and facilitate the correction process. The file data in formatted form is suggested for use in verifying and validating data in the files. The third series of printouts is for programmer use in program check outs and is not discussed here. The list of constant data input printouts is shown in Figure II-7-C-1.

Printout Title	Data Source
80 Column Card Image	Data deck, INC
Constant Data on Sensors	Data file 20
Unattended Ground Sensors	Data file 20
Equipment Item Code for Target Category	Data file 36
SOP for Reporting Hostile Forces	Data file 36
Flow Delay Times	Data file 36
Processing and Decision Delay Times	Data file 36
Unit Size Based on Equipment and Personnel Counts	Data file 36
Target Inferences	Data file 36

Figure II-7-C-1. Printouts of Intelligence and Control Constant Data Input

2. EIGHTY-COLUMN CARD IMAGE. This printout is shown in Figure II-7-C-2. The top line at the far left has the characters 1B indicating that this is card type 1 for Blue forces. At the far right is the number 200 which is data file 20 with format 01. This card is general radar data, and the numbers entered on the top line would be interpreted as follows: equipment item code 84, type of sensor is ground moving; maximum range was 18,280 meters, and so on.

3. CONSTANT DATA ON SENSORS. Figure II-7-C-3 illustrates this printout. The equipment item code of 84 is assigned this sensor with reference number of 201. The minimum and maximum ranges are shown, as was discussed for the card images. The card image data have been formatted and columnar headings applied so that it will be more readily understandable and can be used for validation checks. The follow on card labeled as general radar data continued also has its data displayed in this format. Errors or omissions detected here can then be traced to the appropriate card in the printout illustrated in Figure II-7-C-2 for remedial action.

4. UNATTENDED GROUND SENSORS. The illustration in Figure II-7-C-4 is for the Blue force. Data here can be traced to the card image in Figure II-7-C-2 to take remedial actions if needed. Headings for the various columns are similar to that for the card input format.

5. EQUIPMENT ITEM CODE FOR TARGET CATEGORY. Figure II-7-C-5 illustrates the format of this printout. At the far left the EOH column is the equipment item code. The target category for Blue force is shown in the center column, and that for Red is in the right column.

6. STANDING OPERATING PROCEDURE FOR REPORTING HOSTILE FORCES. These data (Figure II-7-C-6), from data file 36 were originally read from card type 1, ID 3604.

7. FLOW DELAY TIMES. The information from data file 36 on flow delay times is printed in the printout illustrated in Figure II-7-C-7. The data were originally read into data file 36 from card type 1, ID 3601.

8. PROCESSING AND DECISION DELAY TIMES. Figure II-7-C-8 illustrates the printout of data file 36 information. The data were initially read from card type 1, ID 3601. The data from this card were divided between the printout on flow delay times and this printout.

9. UNIT SIZE BASED ON PERSONNEL AND EQUIPMENT COUNTS. The printout is illustrated in Figure II-7-C-9. At the far left of the printout are the abbreviations for the various sized units that may be detected by radar in INC. PLA is for platoon sized unit, PLP is for platoon plus unit, and so on. The minimum and maximum is indicated beneath each of the headings.



1R	844	18200	450	300	130	24	75	30	4	10	2	3	35	25144	2001
1R	822	10000	50	500	44	159	80	27	70	4	5	4	10	2120	2001
1R	898	12000	1000120000	32	2133			85			5	3	35	25	7
1R	592	10000	50	500	44	80	27	70	10	20	5	4	10	2120	2001
1R	554	50000	200	1000	25	83	35	90	20	10	2	3	11	1240	2001
1R	318	16500	500	16500	34	990		90	3	20	5	3	1	1	24
1P	288	35000	1000	35000	44	2100		90	2	180	5	3	90	60	25
2R	844	100	95	50	100	60	100	90	40						2001
2R	822	100	80	60	40	100	70	3	100	85	55				2001
2R	898	100100100100	100	95	40	100	95	40							2001
2R	592	100	80	60	40	100	70	3	100	85	55				2001
2R	554	100	90	85	50	100	60	25	100	90	40				2001
2R	318	100	90	85	50	100	60	25	100	90	40				2001
2R	288	100	90	85	50	100	60	25	100	90	40				2001
3R	841	97	2500	60	4500	2	98	11000	65	15000	8				2001
3R	843	98	12000	60	17000	4	98	11000	60	17000	8				2001
3R	845	97	8000	60	10000	5	97	11000	60	15000	5				2001
3R	847	97	8000	60	10000	10	97	11000	55	15000	5				2001
3R	821	99	2500	97	5000	2	99	4500	65	8000	5				2001
3R	823	99	4500	65	8500	4	99	4500	65	8500	5				2001
3R	825	97	4000	60	7000	5	98	4000	65	7000	5				2001
3R	827	97	4000	50	7000	3	98	4000	65	7000	5				2001
3R	899	99	45000	98	100000	80	2	99	4500	65	8000	5			2001
3R	591	99	2500	97	5000	2	99	4500	65	8000	5				2001
3R	551	99	2500	97	5000	2	98	11000	65	16000	8				2001
3R	319	99	5000	98	16000	80									2001
3R	289	99	10000	98	35000	80									2001
3P	593	99	4500	65	8500	5	4	99	4500	65	8500	5			2001
3R	553	98	12000	60	17000	10	4	98	11000	60	17000	8			2001
3R	595	97	4000	60	7000	5	5	98	4000	65	7000	5			2001
3R	555	97	8000	60	10000	5	5	97	11000	60	16000	5			2001
3R	597	97	4000	50	7000	3	9	98	4000	65	7000	5			2001
3R	557	97	8000	60	10000	10	9	97	11000	55	16000	5			2001
4R	TEST	1	01	1	1002	2003	150	TEST	2	02	1	802	603	90	2001
4P	TEST	3	01	1	2002	3003	100	TEST	4	02	1	902	1103	150	2001

II-7-C-3

Figure II-7-C-2. INC 80-Column Card Image

SENSOR REFERENCE NUMBER	201	EQUIPMENT ITEM	A4	SENSOR TYPE	4	
SENSING RANGE LIMITS IN METERS	---	MAXIMUM	18290	MINIMUM	450	
ROTATION RATE IN REV/S / MIN	0					
SCAN RATE IN MILS / SEC	23					
RADAR BEAM --- HORIZONTAL WIDTH IN MILS	179					
DEPTH IN METERS	900					
TIME NEEDED IN MINUTES -- POWER UP	25	POWER DOWN	35			
MEDIAN TIME TO TRACK A TARGET AND GENERATE INFORMATION IN SECONDS	144					
PERCENT DOWN TIME	2					
DEFLECTION ERROR IN MILS	3					
RANGE ERROR AT TARGET LOCATION IN METERS	10					
RANGES IN METERS						
	KNEE	ANGLE	PROBABILITY OF COLLECTION			
			AT GIVEN RANGES			
			MIN	KNEE	ANGLE	MAX
PEOPLE	2500	4500	97	50	0	0
WHEELED VEHICLES	11000	16000	98	65	8	0
TANKS	12000	17000	98	60	10	0
APCS	11000	17000	98	60	8	0
ARTY TUBES	8000	10000	97	50	5	0
ARTY MISSILES	11000	16000	97	60	5	0
ADA GUNS	8000	10000	97	50	10	0
ADA MISSILES	11000	16000	97	55	5	0
PRECIPITATION FACTORS --	NONE	100	LIGHT	90	HEAVY	40
ECM FACTORS	NONE	100	SOME	60	MORE	25
FORESTATION FACTORS --	NONE	100	SPARSE	90	MODERATE	85
FALSE ALARM PROBABILITY						DENSE
RESULTS RELIABILITY FACTOR	97					50

Figure II-7-C-3. Constant Data on Sensors

UNATTENDED GROUND SENSORS FOR BLUE						
SENSOR TYPE	TARGET - RANGE	TARGET - RANGE	TARGET - RANGE	TARGET - RANGE	FALSE ALARM PERCENT	
1	1	100	2	200	3	150
2	1	90	2	50	3	90
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0
11	0	0	0	0	0	0
12	0	0	0	0	0	0
13	0	0	0	0	0	0
14	0	0	0	0	0	0
15	0	0	0	0	0	0
16	0	0	0	0	0	0
17	0	0	0	0	0	0
18	0	0	0	0	0	0
19	0	0	0	0	0	0
20	0	0	0	0	0	0

II-7-C-5

Figure II-7-C-4. Unattended Ground Sensors for Blue Force

# CODE LETTER BY EOH

EOH BLUE PFD

1	0	0
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0
10	0	0
11	0	0
12	0	0
13	0	0
14	0	0
15	0	0
16	0	0
17	0	0
18	0	0
19	0	0
20	C	F
21	0	0
22	0	0
23	C	J
24	0	P
25	0	0
26	0	J
27	G	P
28	0	0
29	E	0
30	0	I
31	0	0
32	E	T
33	0	0
34	0	0
35	0	0
36	0	0
37	0	F
38	0	P
39	I	0
40	0	0
41	K	0
42	0	F
43	K	0
44	0	F

Figure II-7-C-5. Equipment Item Code for Target Category

12/00/71 INTELLIGENCE AND CONTROL 15/35/42

BATTALION											
SIZE	TYPE	ACT	PERS	TANKS	APC	TUBES	WEAPONS	SIZE	TYPE	ACT	PERS
DIVISION	5	0000	0000	0000	21	10	10	DIVISION	5	0000	0000
BATTALION	3	0000	0000	0000	4	4	5	BATTALION	3	0000	0000
DIV ARV	3	0000	0000	0000	4	4	10	DIV ARV	3	0000	0000
DIP Super	3	0000	0000	0000	4	4	5	DIP Super	3	0000	0000
BATTALION											
DIVISION	5	0000	0000	0000	21	10	10	DIVISION	5	0000	0000
BATTALION	3	0000	0000	0000	4	4	5	BATTALION	3	0000	0000
DIV ARV	3	0000	0000	0000	4	4	10	DIV ARV	3	0000	0000
DIP Super	3	0000	0000	0000	4	4	5	DIP Super	3	0000	0000
BATTALION											
DIVISION	5	0000	0000	0000	21	10	10	DIVISION	5	0000	0000
BATTALION	3	0000	0000	0000	4	4	5	BATTALION	3	0000	0000
DIV ARV	3	0000	0000	0000	4	4	10	DIV ARV	3	0000	0000
DIP Super	3	0000	0000	0000	4	4	5	DIP Super	3	0000	0000
BATTALION											
DIVISION	5	0000	0000	0000	21	10	10	DIVISION	5	0000	0000
BATTALION	3	0000	0000	0000	4	4	5	BATTALION	3	0000	0000
DIV ARV	3	0000	0000	0000	4	4	10	DIV ARV	3	0000	0000
DIP Super	3	0000	0000	0000	4	4	5	DIP Super	3	0000	0000

Figure II-7-C-6. Standing Operating Procedure for Reporting Hostile Forces

12/09/71 INTELLIGENCE AND CONTROL 15/35/42

FLOW DELAY TIMES

	DIVISION	BRIGADE	BATTALION	DIV-ARTY	DIR-SUPT	DIVISION	BRIGADE	BATTALION	DIV-ARTY	DIR-SUPT
DIVISION	0	3100	2600	0	0	DIVISION	0	3100	2600	0
BRIGADE	730	960	2400	0	0	BRIGADE	730	960	2400	0
BATTALION	0	960	620	0	0	BATTALION	0	960	620	0
DIV-ARTY	730	960	0	0	200	DIV-ARTY	730	960	0	200
DIR-SUPT	0	960	620	100	0	DIR-SUPT	0	960	620	100
DELAY TIME - NAFS - CAS - ARTY	1000	2000	100	BLUE						
DELAY TIME - NAFS - CAS - ARTY	0	2000	100	RED						

II-7-C-8

Figure II-7-C-7. Flow Delay Times

[illegible]

Figure II-7-C-8. Processing and Decision Delay Times

II-7-C-9

12/09/71 INTELLIGENCE AND CONTROL 15/35/42

BLUE

	INFANTRY		ARMOR		MECH		RTF		TANK		RTF		APC		TURS		MSSL		GUNS		AJA		MSSL		ACFT		VEH	
	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX
PLA	0	0	0	0	2	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
PLP	0	0	0	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
CO	0	0	0	3	9	3	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
COP	0	0	0	10	20	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
WH	0	0	0	21	31	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SNP	0	0	0	22	61	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RDE	0	0	0	64	95	59	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure II-7-C-9. Unit Size Based on Equipment and Personnel Counts

II-7-C-10



10. TARGET INFERENCES. The printout for target inferences is illustrated in Figure II-7-C-10. The Intelligence and Control logic first asks for the size of the unit based on count of personnel and equipment. When this has been obtained, then the unit size is further examined in terms of the normal interval between the senior and subordinate units. It is this normal interval that is expressed in meters here. The information from the file was initially read in from card types 1 and 2, ID 3603.

12/09/71

INTELLIGENCE AND CONTROL

15/35/42

BLUE

INFANTRY	APMOR	MECH	PTF	TANK	PTF	APC	TUGES	MSSL	GUNS	AGA	MSSL	ACFT	VEH
PLA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLP	0	0	0	0	0	0	0	0	0	0	0	0	0
CO	0	600	600	0	0	0	300	300	300	300	300	300	300
BAR	0	1300	1300	0	0	0	500	1300	1300	1300	500	500	500
ANP	0	5000	5000	0	0	0	0	0	0	0	0	0	0
AOE	0	8000	8000	0	0	0	0	0	0	0	0	0	0

Figure II-7-C-10. Target Inferences

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR INTELLIGENCE AND CONTROL

(AVAILABLE UNDER SEPARATE COVER)

NOTES

II-7-D-2

## CHAPTER 8

### GROUND COMBAT DATA INPUT PROGRAMS

1. INTRODUCTION. The Ground Combat data input programs create and load the data file used by the Ground Combat Model. The data include tables describing weapon characteristics, sensor distribution, and equipment and personnel losses for attackers and defenders in a ground combat situation.

#### 2. ROUTINES:

a. General. The Ground Combat data input programs are composed of the routines described as follows.

b. Routine GCMLD. This routine has the responsibility of reading and editing the data cards, storing the data on the file, and calling the file dump routine DUMP39. It is supported by utility routines AROTE, LINEAR, SPP, and FLIP.

c. Routines AROTE, LINEAR, SPP, and FLIP. Routine AROTE determines the presented area of a double box combination after being rotated through a specific angle. LINEAR performs a linear least squares fit to a set of points, and determines the slope and Y intercept of the resulting line segment. SPP determines the coefficients a and b in the expression  $f(r,t = 10) = ae^{-br}$  and FLIP initializes the second record of data file 39.

d. Routine DUMP39. This routine is used to provide a formatted dump of the data file.

3. FILES. The following data file is loaded by these routines.

. Ground combat data - data file 39

NOTES

II-8-2

## APPENDIX A

### GROUND COMBAT DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for Ground Combat are documented in Appendix A, Ground Combat Model Input Requirements, to Chapter 7 of the Period Processor (Section IV).

NOTES

II-8-A-2



## APPENDIX B

### GROUND COMBAT DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. Data for the Ground Combat Model are loaded on data file 39. Loading is accomplished by routine GCMLD and supporting routines AROTE, LINEAR, SPP, FLIP, and DUMP39.

#### 2. ROUTINE GCMLD:

a. Purpose. Routine GCMLD identifies errors in the constant data and prints a diagnostic message if an error is found. The routine branches to the proper logic for various data types, and stores each data item in the proper location after data manipulation, if required.

#### b. Input Variables:

(1) Data file 39 variables.

(2) Other variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ICARD(20)	Card	Data card image.

#### c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
IDAT(6000)	DF39	Array containing data to be placed on data file 39.

#### d. Logical Flow (Figure II-8-B-1):

(1) Block 1 and 1A. Obtain the file name table; create data file 39 with two records containing 14,258 words each and data file 21 with 105 records containing 256 words each.

(2) Block 2. Sensor type 1, unaided visual detection, which is not defined by input, is designated as applicable to both day and night detection.

(3) Block L100. Each data card is read as alphanumeric characters.

(4) Block 3. Part of the printed output of GCMLD is a list of card images. Each card image is printed at this time.

(5) Block 4. If card identification does not equal 9999, control goes to block L102.

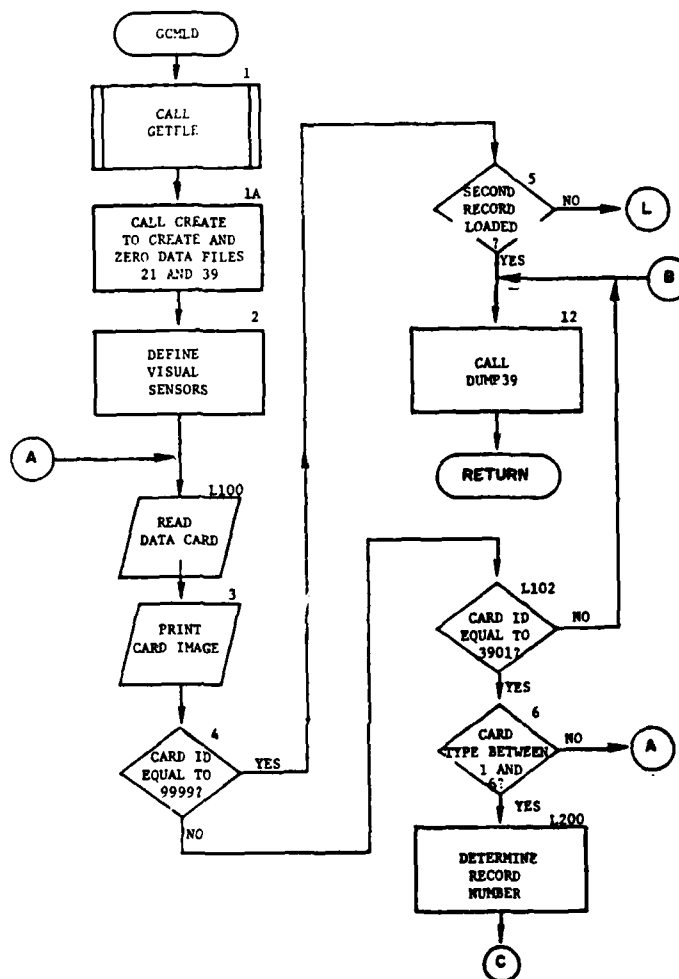


Figure II-8-B-1. Routine GCMLD (Continued on Next Page)

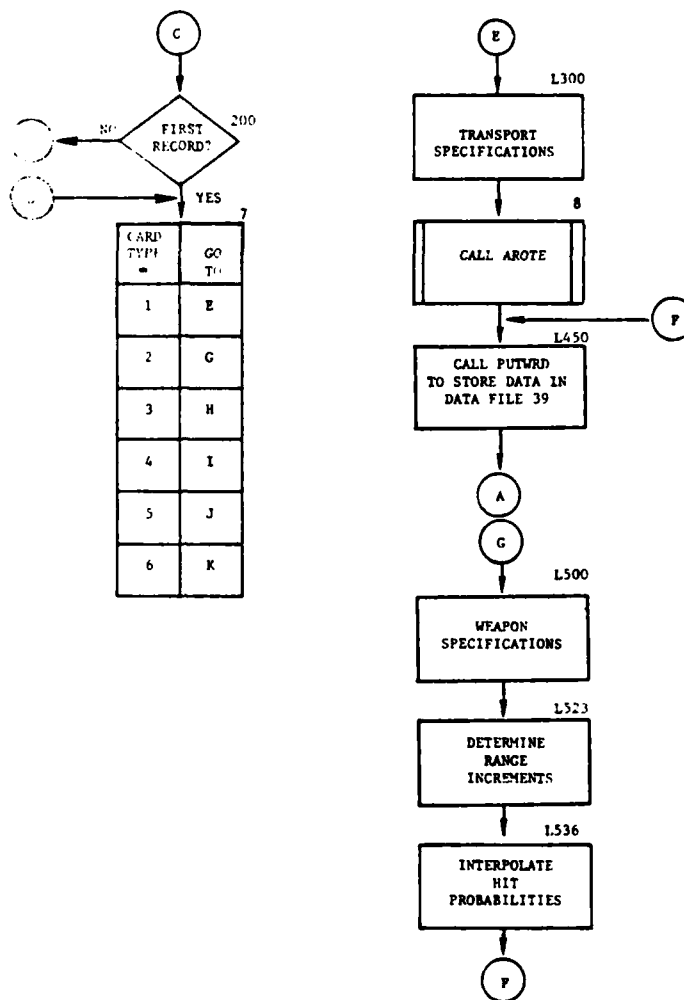


Figure II-8-B-1. Routine GCMLD (Continued)

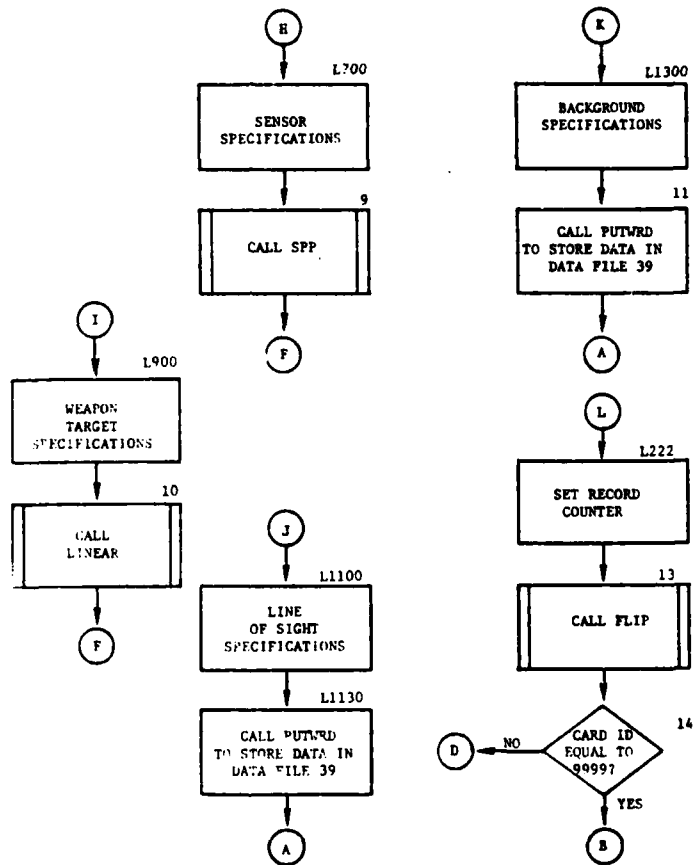


Figure II-8-B-1. Routine GCMLD (Concluded)

(6) Block 5. If the second record has not yet been loaded, control goes to block L222; otherwise, to block 12.

(7) Block 12. This block completes input. Complete the output by calling DUMP39; control returns to calling routine.

(8) Block L102. Any card identification other than 3901 will cause processing to be terminated.

(9) Block 6. If the card type is less than one or greater than six a diagnostic message is printed and another card is read.

(10) Block L200. If the data designate Blue as the defender or Red as the attacker (data for the second record), control goes to block L222; otherwise, it continues to block 7.

(11) Block 7. Control branches to the logic for the particular card type.

(12) Blocks L300, 8, and L450. Decode the data according to transport system format specification, compute the presented area following a 30-degree rotation by a call to AROTE, and store the data in the proper record of data file 39. Control then returns to block L100.

(13) Blocks L500, L523, and L536. Decode the data according to weapon system format specification. The range values in this set of data are scanned to determine the smallest (minimum range) and largest (maximum range). The resulting range interval is subdivided into five equal increments and a linear interpolation is performed on the hit probability data to determine the value corresponding to each range increment. Control is transferred to block L450.

(14) Blocks L700 and 9. Decode the data according to sensor format specification and determine the performance parameters required by the Ground Combat Model calling SPP. Control is transferred to block L450.

(15) Blocks L900 and 10. Decode the data according to weapon-target format specification. Using a least squares linear fit by calling LINEAR to determine the slope and Y intercept of the  $P_K/H$  line segment. Control is then transferred to block L450.

(16) Blocks L1300 and L1130. Decode the data according to line of sight format specification and store it in both records of data file 39. Transfer control to block L100.

(17) Blocks L1300 and 11. Decode the data according to background format specification and store it in both records of data file 39. Transfer control to block L100.

(18) Blocks L222 and 13. The data for one record is complete. By calling FLIP, interchange all attacker and defender data to initialize the next record.

(19) Block 14. If the card ID is 9999, the second record is the reverse of the first. Store it and terminate; otherwise, some data in the second record require redefinition. Go to block 7.

### 3. ROUTINE AROTE:

a. Purpose. AROTE determines the presented area of a double box combination after being rotated through a specified angle.

#### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IDIM(6)	Call	Length, width, and height of two boxes.
IDEQ	Call	Angle, in degrees, through which the box combination is to be rotated.

#### c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
ANS	Call	Presented area, following rotation.

d. Processing Description. AROTE multiplies each horizontal dimension of each box by the appropriate sine or cosine of the angle, and takes the product of these resulting projected dimensions with the height. The presented area is determined by summing the area of each box.

### 4. ROUTINE LINEAR:

a. Purpose. LINEAR performs a linear least squares fit to a set of points, and determines the slope and Y intercept of the resulting line segment.

#### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
N	Call	Number of coordinate sets in the array of input points.
XY(6,3)	Call	Array of input points.

#### c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
A	Call	Y intercept.
B	Call	Slope.

d. Processing Description. LINEAR calculates the slope and Y intercept using the following equations:

$$\text{Slope} = \frac{\sum_{i=1}^N X_i Y_i - \frac{1}{N} \sum_{i=1}^N X_i \sum_{i=1}^N Y_i}{\sum_{i=1}^N (X_i)^2 - \frac{1}{N} \left( \sum_{i=1}^N X_i \right)^2} \quad \text{II-8-B-1}$$

$$\text{Y intercept} = \frac{\sum_{i=1}^N Y_i - (\text{Slope}) \sum_{i=1}^N X_i}{N} \quad \text{II-8-B-2}$$

#### 5. ROUTINE SPP:

a. Purpose. SPP determines the coefficients a and b in the expression  $f(r, t = 10) = ae^{-br}$  given the value of the function for two values of r at arbitrary t assuming the relation  $f(r, t) = 1 - [1 - f(r, t = 10)]^{t/10}$ .

##### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IHOLD(10)	Call	Two values of r with the value of the function evaluated at each r and the value of t.
NSENTD	Call	Pointer to the location in DSPP where output values are to be stored.

##### c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
DSPP(10,4)	Call	Output array giving, for each value of NSENTD, the calculated values of a and b as well as the values of r for which the function $ae^{-br}$ attains the values of 0.9999 and 0.0001.

d. Processing Description. SPP evaluates the expressions:

$$b = \frac{\ln \left\{ 1 - [1 - f(r_2, t)]^{10/t} \right\} - \ln \left\{ 1 - [1 - f(r_1, t)]^{10/t} \right\}}{r_1 - r_2} \quad \text{II-8-B-3}$$

II-8-B-7

$$a = e^{br_1} [1 - \{1 - f(r_1, t)\}^{10/t}]$$

II-8-B-4

#### 6. ROUTINE FLIP:

a. Purpose. FLIP initializes the second record of data file 39.

b. Input Variables:

<u>Name</u> ,	<u>Source</u>	<u>Contents</u>
I(10)	Call	One of two arrays to be interchanged.
J(10)	Call	One of two arrays to be interchanged.
K	Call	Number of words in the pair of arrays to be interchanged.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
I(10)	Call	Interchanged array.
J(10)	Call	Interchanged array.

d. Processing Description. FLIP interchanges the first K words of arrays I and J.

#### 7. ROUTINE DUMP39:

a. Purpose. DUMP39 provides a formatted output capability to the loading program. It is designed to display the contents of both records of data file 39 in tabular form.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
FILE39(6000)	DF39	Ground Combat Model constant data.

c. Output Variables. Individual data file 39 variables.

d. Logical Flow (Figure II-8-B-2):

(1) Block 1. The record to be output is obtained from data file 39.

(2) Blocks 2 and L100. Environmental data are printed only in conjunction with the first record. The data are identical with that in the second record and are not repeated. Included are the line of sight parameter, background reflectance, and sky/ground ratio tables.

II-8-B-8



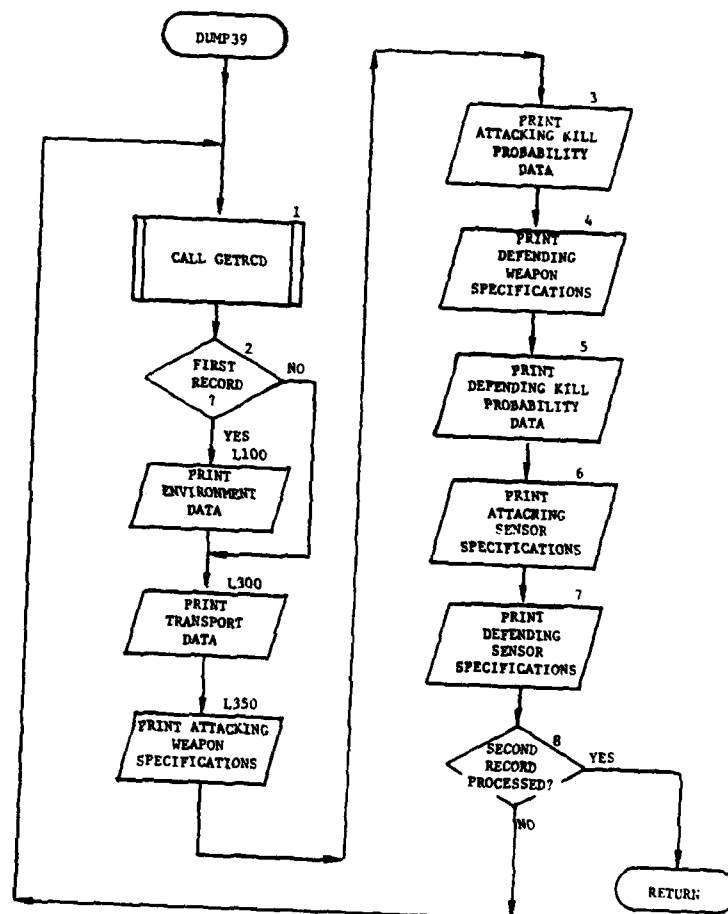


Figure II-8-B-2. Routine DUMP39

II-8-B-9

(3) Block L300. Print transport specifications for each force including transport area, type, reflectance, probability of being pinpointed, and the number of associated personnel.

(4) Block L350. Print attacking weapon specifications including range capabilities, delivery times, firing rates, NATO hit probabilities, and target priorities.

(5) Block 3. Print slopes and Y intercepts of kill probability line segments for attacking weapons against defending targets.

(6) Blocks 4 and 5. Print data for defending weapons specified in blocks L350 and 3.

(7) Blocks 6 and 7. Print sensor specifications for each force including sensor utilization and sensor performance parameters.

(8) Block 8. If both records have not been processed, transfer control to block 1 for the second record; otherwise, return control to calling routine.

## APPENDIX C

### GROUND COMBAT DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. The input of Ground Combat constant data in the DIVWAG Model results in the generation of a series of output reports. Three types of output reports are printed: the 80-column card image, the formatted data from data file 39, and the data file 39 dump. The file printouts are listed in Figure II-8-C-1.

Printout Titles	Data Source File
80-Column Card Image	Ground Combat Data Deck
Line of Sight	Data file 39
Targets and Pinpoint Probabilities	Data file 39
Weapons and Specifications	Data file 39
Weapon/Target Intercept	Data file 39
Sensor Distribution and Performance	Data file 39

Figure II-8-C-1. Ground Combat Printouts for  
Constant Data Input

2. EIGHTY-COLUMN CARD IMAGE. This format is illustrated in Figure II-8-C-2. The top row, far left, has the characters 1BA. The 1 specifies the card type; The B specifies the data are for a Blue transport; and the A specifies the data pertain to an attacking Blue unit. At the far right of the figure the number 3901 appears, indicating that the data are for data file 39 and format 01. It also is a positive identifier for this specific card. The remaining information on this card is identifiable by tracing the headings on the card and relating the data entries to these topics.

3. LINE OF SIGHT. An illustration of this printout is shown in Figure II-8-C-3. Four tables are presented in this printout. The first, at the top, is line of sight parameters with no forest, and line of sight parameters for terrain with forest. The second is located in the center of the figure and entitled background reflectance. Headings of these columns are similar to those found on the card format. The third table is labeled sky-ground

II-8-C-2

Figure II-8-C-2. Ground Combat 80-Column Card Image

LINE OF SIGHT

TERRAIN	NO FOREST		FOREST	
	POSTURE 1	POSTURE 2	POSTURE 1	POSTURE 2
	PERSONNEL	VEHICLES	PERSONNEL	VEHICLES
GOOD	300.	450.	500.	600.
BAD	500.	300.	350.	350.
				325.
				175.

BACKGROUND REFLECTANCE

SEASON	BARREN	SPARSE	PLENTIFUL
SUMMER	.184	.184	.755
AUTUMN	.755	.755	.755
SNOW	.085	.135	.755

SKY-GROUND RATIO

TOWARD THE SUN	SUN OVERHEAD	AWAY FROM THE SUN
4.0	3.0	2.0

MINIMUM FRONT TO FRONT SEPERATION

50 METERS

Figure II-8-C-3. Line of Sight

ratio and contains the values for the three illumination conditions. The fourth is the minimum front-to-front separation of the opposing forces. The data shown in this printout, taken from data file 39, were originally read in by the cards recognized by card types 5 and 6, ID 3901.

4. TARGET AND PINPOINT PROBABILITIES. The third set of printout formats is that of target and pinpoint probabilities as illustrated in Figure II-8-C-4. There are three major divisions to this printout format, each of which has two tables. The upper third of the printout has the target data for both attacking and defending. In each table at the top there is the column (second from the left) labeled presented area. This is a computed figure from the information entered into card type 1, ID 3901, titled Transport Specifications. The information in the column headed presented area is the area in square feet of the target rotated through 30 degrees, and the 30-degree area is that presented as a target and computed for this column. The pinpoint probability and mobility class rate tables are also taken from card type 1, ID 3901.

5. WEAPON SYSTEM SPECIFICATIONS. The fourth set of printouts is weapon system specifications illustrated in Figure II-8-C-5. This format has three major tables. At the top, labeled link table, is a list of the ammunition (by item code) carried by each transport (by item code). In the center is the table with weapon performance data with each weapon's range limitations; maximum rate of fire in rounds per minute; the average time in seconds to aim, fire, and deliver a round to the midpoint of its effective range; and hit probabilities against a standard NATO target at six range values. At the bottom of the figure is the third table, labeled priority (ammo to drop) weapon to target. At the far left of this table is the equipment item code of the weapon. Across the top is the equipment item code for each of the hostile targets (transports). Beneath that listing is the priority given that target for the specific weapon. In parentheses following the target priority is the percentage of on-board basic load of ammunition which, when reached, will cause that target to be dropped from the priority list. Once dropped from the priority list, that target will not be fired on again by that weapon until the Combat Service Support Model is able to replenish the ammunition.

6. WEAPON TARGET KILL PROBABILITY. This printout is shown in Figure II-8-C-6. The data included in these tables are computed from the information originally entered in card type 2, weapon systems specifications, ID 3901. The far left, ammo EOH, is the equipment item code for the specific weapon. Target, in the center top of the figure, is the hostile force target (transport) item code. Beneath that equipment item code on the first line is the intercept (see second column from left). The intercept is the probability of a kill given a hit at zero range. The second line, labeled slope, is the tangent of the angle made by a straight line, defined by the intercept and the probability of a kill given a hit at the maximum range and the abscissa, representing the range from firer to target.

ATTACKING TARGET				DEFENDING TARGET			
EOH	PRESENTED AREA	NUMBER OF PERSONNEL	REFLECTANCE	EOH	PRESENTED AREA	NUMBER OF PERSONNEL	REFLECTANCE
31	6.2	1.00	.100	35	117.89	4.0	.120
34	33.8E	4.00	.120	36	156.52	3.0	.120
35	10.5L	4.00	.120	161	138.56	2.0	.120
37	195.10	4.00	.120	160	109.54	5.0	.120
61	121.89	4.00	.120	32	109.54	4.0	.120
45	8.26	1.00	.100	23	8.26	1.0	.100
185	104.50	1.00	.100	25	8.20	3.0	.100
186	131.00	3.00	.120	27	16.79	3.0	.100

ATTACKING TARGET		DEFENDING TARGET	
EOM	PROBABILITY	EOM	PROBABILITY
31	0.669	35	0.009
34	0.336	36	0.000
35	0.520	161	0.000
37	0.306	160	0.009
41	0.626	32	0.000
45	0.600	23	0.000
85	0.530	25	0.000
166	0.530	27	0.000

ATTACKING TARGET		DEFENDING TARGET	
EOH	RATE	EOH	RATE
31	133.336	35	133.336
34	133.336	36	133.336
35	133.336	161	133.336
37	133.336	160	133.336
41	133.336	32	133.336
45	133.336	23	133.336
165	133.336	25	133.336
166	133.336	27	133.336

II-8. C-5

AMMUNITION CODE	32	36	36	38	39	40	42	43	9	44	0	0
TRANSPORT CODE	31	34	35	37	37	37	41	41	185	186	45	0

ANNO LEON	MIN RANGE	MAX RANGE	RATE	TIME	HIT PROBABILITIES
32	70-800	100-1300	.500	13.292	.960
35	65-800	300-1300	.500	13.862	.960
36	65-800	300-1300	.500	13.862	.978
37	65-800	300-1300	.500	13.862	.978
38	100-800	300-1300	.500	16.562	.960
39	100-800	300-1300	.500	22.150	.980
40	100-800	300-1300	.500	22.150	.980
41	100-800	300-1300	.500	22.150	.980
42	100-800	300-1300	.500	16.600	.922
43	100-800	300-1300	.500	16.600	.922
44	100-800	300-1300	.500	16.600	.922
45	100-800	300-1300	.500	16.600	.922
46	100-800	300-1300	.500	16.600	.922
47	100-800	300-1300	.500	16.600	.922
48	100-800	300-1300	.500	16.600	.922
49	100-800	300-1300	.500	16.600	.922
50	100-800	300-1300	.500	16.600	.922
51	100-800	300-1300	.500	16.600	.922
52	100-800	300-1300	.500	16.600	.922
53	100-800	300-1300	.500	16.600	.922
54	100-800	300-1300	.500	16.600	.922
55	100-800	300-1300	.500	16.600	.922
56	100-800	300-1300	.500	16.600	.922
57	100-800	300-1300	.500	16.600	.922
58	100-800	300-1300	.500	16.600	.922
59	100-800	300-1300	.500	16.600	.922
60	100-800	300-1300	.500	16.600	.922
61	100-800	300-1300	.500	16.600	.922
62	100-800	300-1300	.500	16.600	.922
63	100-800	300-1300	.500	16.600	.922
64	100-800	300-1300	.500	16.600	.922
65	100-800	300-1300	.500	16.600	.922
66	100-800	300-1300	.500	16.600	.922
67	100-800	300-1300	.500	16.600	.922
68	100-800	300-1300	.500	16.600	.922
69	100-800	300-1300	.500	16.600	.922
70	100-800	300-1300	.500	16.600	.922
71	100-800	300-1300	.500	16.600	.922
72	100-800	300-1300	.500	16.600	.922
73	100-800	300-1300	.500	16.600	.922
74	100-800	300-1300	.500	16.600	.922
75	100-800	300-1300	.500	16.600	.922
76	100-800	300-1300	.500	16.600	.922
77	100-800	300-1300	.500	16.600	.922
78	100-800	300-1300	.500	16.600	.922
79	100-800	300-1300	.500	16.600	.922
80	100-800	300-1300	.500	16.600	.922
81	100-800	300-1300	.500	16.600	.922
82	100-800	300-1300	.500	16.600	.922
83	100-800	300-1300	.500	16.600	.922
84	100-800	300-1300	.500	16.600	.922
85	100-800	300-1300	.500	16.600	.922
86	100-800	300-1300	.500	16.600	.922
87	100-800	300-1300	.500	16.600	.922
88	100-800	300-1300	.500	16.600	.922
89	100-800	300-1300	.500	16.600	.922
90	100-800	300-1300	.500	16.600	.922
91	100-800	300-1300	.500	16.600	.922
92	100-800	300-1300	.500	16.600	.922
93	100-800	300-1300	.500	16.600	.922
94	100-800	300-1300	.500	16.600	.922
95	100-800	300-1300	.500	16.600	.922
96	100-800	300-1300	.500	16.600	.922

	35	36	161	160	TARGET EDH	32	23	25	27
32	11	21	31	41	51	01	01	01	01
36	11	21	31	41	51	01	01	01	01
36	11	21	31	41	51	01	01	01	01
38	11	21	31	41	51	01	01	01	01
39	11	21	31	41	51	01	01	01	01
43	11	21	31	41	51	01	01	01	01
42	11	21	31	41	51	01	01	01	01
43	11	21	31	41	51	01	01	01	01
9	11	21	31	41	51	01	01	01	01
9	11	21	31	41	51	01	01	01	01
44	11	21	31	41	51	01	01	01	01
0	11	21	31	41	51	01	01	01	01
0	11	21	31	41	51	01	01	01	01
0	11	21	31	41	51	01	01	01	01
3	11	21	31	41	51	01	01	01	01
3	11	21	31	41	51	01	01	01	01



# ATTACKING AMMUNITION

AMMO EOW	35	35	101	TARGET 160	32	23	25	27
32 INTERCEPT SLOPE	.44000 0.00000	.44000 0.00000	.06000 0.00000	.65000 0.00000	.66000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000
36 INTERCEPT SLOPE	.48000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000
36 INTERCEPT SLOPE	.48000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000
38 INTERCEPT SLOPE	.46000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000
39 INTERCEPT SLOPE	.496427 0.00000	.70000 0.00000	.70000 0.00000	.70000 0.00000	.51000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000
40 INTERCEPT SLOPE	.46000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000
42 INTERCEPT SLOPE	.46000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000
43 INTERCEPT SLOPE	.49000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000
9 INTERCEPT SLOPE	.46000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000
9 INTERCEPT SLOPE	.46000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	.95000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000
44 INTERCEPT SLOPE	.22000 0.00000	.22000 0.00000	.11000 0.00000	.11000 0.00000	.11000 0.00000	0.00000 0.00000	0.00000 0.00000	0.00000 0.00000

II-8-C-7

Figure II-8-C-6. Weapon/Target Kill Probabilities

7. SENSOR DISTRIBUTION AND PERFORMANCE. This printout is shown in Figure II-8-C-7. This format has two major tables, one of which has the distribution of sensors among the various transports, stating the amount of time that they are used either while aboard that transport or while dismounted but supporting that transport. The sensor is identified at the far left under the column headed sensor EOH, which is the equipment item code. The transport is identified with its equipment item code listed across the first row of the top table. Beneath that figure is the listing of the times that the sensor is operated for that transport. The data are extracted from card type 3, sensor specifications, ID 3901. The sensor performance data are calculated from information provided by this card type and are used to calculate the probability of detection for each sensor.

[illegible][illegible]

NOTES

II-8-C-10

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR GROUND COMBAT

(AVAILABLE UNDER SEPARATE COVER)

NOTES

II-8-D-2

## CHAPTER 9

### AREA FIRE DATA INPUT PROGRAMS

1. INTRODUCTION. The Area Fire data input program creates and loads the files containing the munitions lethal areas and personnel protection data required by the Area Fire Model. It also loads certain weapons parameters on data file 25 where they are used by the TACFIRE Model.

#### 2. ROUTINES:

a. General. The Area Fire data input programs consist of the routines described in the following paragraphs.

b. Routine AFMLD. This routine controls the loading of the two data files required by the Area Fire Model. It creates the data files, and then calls routines LOAD29, DUMP29, LOAD30 and DUMP30 in that sequence.

c. Routine LOAD29. This routine reads and edits the munition characteristics data and stores it in the correct position on the files.

d. Routine DUMP29. This routine gives a formatted dump of data file 29 and the portion of data file 25 loaded by the input routines.

e. Routine LOAD29. This routine is responsible for the reading and editing of the personnel protection data and for loading it correctly in the data file.

f. Routine DUMP30. Routine DUMP30 provides a tabular dump of the personnel protection data file.

3. FILES. The files loaded by the Area Fire data input programs are as follows.

. Munitions lethal areas	-	data file 29
. Personnel protection	-	data file 30

NOTES

II-9-2



## APPENDIX A

### AREA FIRE DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for area fire are documented in Appendix A, Area Fire Model Input Requirements, to Chapter 8 of the Period Processor (Section IV).

NOTES

II-9-A-2

## APPENDIX B

### AREA FIRE DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. The Area Fire Model's input data are loaded into data files 20 and 30, by the control routine AFMLD and associated routines LOAD29, LOAD30, DUMP29, and DUMP30. The unit dimension, geometry, and distribution data used by the Area Fire Model are loaded into data file 28 by the routines documented in Chapter 6, Appendix B, Unit Geometry Data Input Program Descriptions.

2. ROUTINE AFMLD:

a. Purpose. AFMLD controls loading of data files required by the Area Fire Model; (i.e., data file 29, which is the munitions lethal areas file, and data file 30, which is the personnel protection file. AFMLD also loads certain weapon parameters onto data file 25 to be used by the TACFIRE Model. Input data come from punched cards. AFMLD relies on two load routines to load the data, LOAD29 and LOAD30, and two dump routines to print the data, DUMP29 and DUMP30.

b. Input Variables: None.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
IDUM(1200)	DF25, DF29 and DF30	Buffer for initializing to zero.

d. Logical Flow (Figure II-9-B-1):

(1) Block 1. The file name table (DIVWAG file directory) is brought in from the master data file by the standard call: CALL GETFLE (4HKEYS, IFNT, IER).

(2) Blocks 2, 3, and 4. Data files 25, 29, and 30 are created if they do not yet exist by use of the CREATE routine. Data file 25 consists of one record which is 27092 words in length, and portions are loaded by two different programs, LOAD29 and TACLD. Data file 29 consists of 72 records of 216 words per record and is loaded entirely by LOAD29. Data file 30 consists of 21 records of 2149 words per record. The first two are loaded by routine LOAD30 and the remainder are loaded by the Nuclear Model data loading routine.

(3) Blocks 5, L300, and L500. These portions of data files 25, 29, and 30 which are to be loaded (i.e., all of data files 29 and 30 and words 7605-8396 and 26877-27092 of data file 25) are initialized to zero.

(4) Blocks 6, 7, and 8. Routine LOAD29 is called to load the munition characteristics data onto data files 25 and 29. NERR is returned as the count of serious errors which were found in the data. Routine DUMP29 is called to print the data from data file 29 if no serious errors were located by LOAD29.

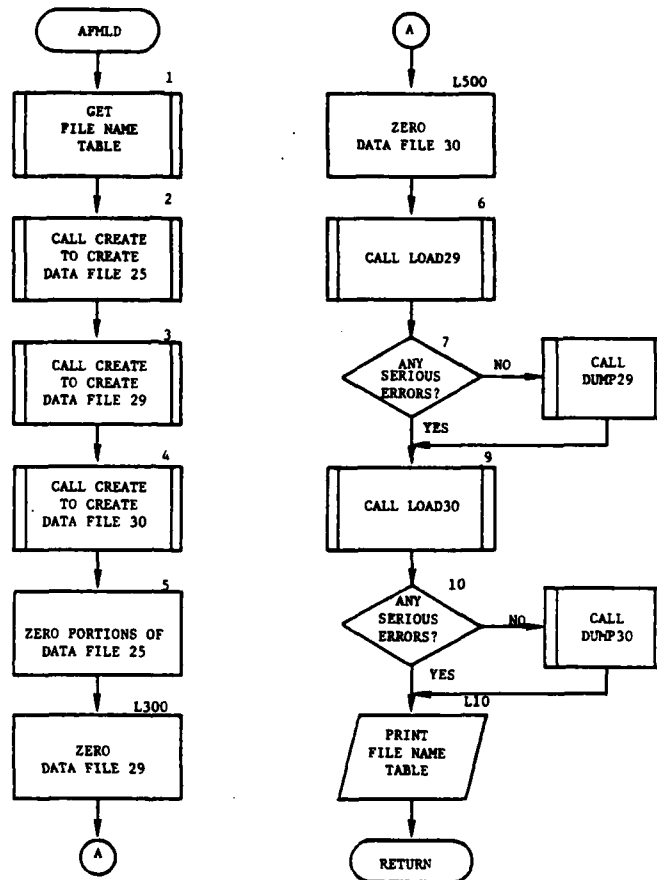


Figure II-9-B-1. Routine AFMLD

II-9-B-2

(5) Blocks 9, 10, and 11. Routine LOAD30 is called to load the personnel protection data onto data file 30. NERR is returned as the count of serious errors which were found in the data. Routine DUMP30 is called to print the data from data file 30 if no serious errors were located by LOAD30.

(6) block L10. The file name table (IFNT) is printed to allow verification of the sizes of data files 25, 29, and 30.

### 3. ROUTINE LOAD29:

a. Purpose. LOAD29 reads the munition characteristics data required by the Area Fire and TACFIRE Models from cards and loads the data on data files 29 and 25.

#### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
KARD(20)	Card	Data card image buffer.
IFILE	Card	File number (29).
IFORM	Card	Card format number (1-4).
ISEQ	Card	Card image sequence number.
ISIDE	Card	Force designator (B or R).
IWMUNX	Card	Weapon/munition combination index (1-36).
IWEOHX	Card	Weapon item code.
IMEOHX	Card	Munition item code.
LAPERS(3,2)	Card	Lethal areas to personnel in three postures (standing, prone, foxhole) and two environments (unforested and forested).
IEOH(7)	Card	List of seven equipment item codes.
LAREA(7)	Card	List of seven lethal areas for equipment items given by IEOH.
MAXRNG	Card	Maximum range of munition.
MINRNG	Card	Minimum range of munition.
ETYPE	Card	Error type code: S=Sigma, C=CEP.
SIGTAB(6)	Card	Table of round dispersion errors at six range points between MAXRNG and MINRNG.

<u>Name</u>	<u>Source</u>	<u>Contents</u>
DTV1	Card	Time to fire first volley.
DTVMR	Card	Time to fire each successive volley at maximum rate of fire.
DTVSR	Card	Time to fire each successive volley at sustained rate of fire.
NRNDMR	Card	Maximum number of rounds that can be fired at maximum rate of fire.
EFFRAD	Card	Effects radius.
IANGF	Card	Angle of fire of the weapon.
LENG	Card	Length of the projectile.
ICAL	Card	Caliber of the projectile.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
NERR	Call	Number of serious data errors detected.
ISEQ	Print	Card image sequence number.
KARD(20)	Print	Data card image buffer.
MUNDAT(216)	DF29	Munition data, data file 29 record.
IWEOHX		MUNDAT(1). Weapon item code.
IMEOHX		MUNDAT(2). Munition item code.
LAPERS(3,2)		MUNDAT(3). Lethal areas for personnel in three postures (standing, prone, foxhole) and two environments (unforested and forested).
LAEOH(200)		MUNDAT(9). Lethal areas for 200 equipment items.
MAXRNG		MUNDAT(209). Maximum range of munition.
MINRNG		MUNDAT(210). Minimum range of munition.
SIGTAB(6)		MUNDAT(211). Round dispersion errors at six range points between maximum range and minimum range.

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
WMPARM(36,11,2) DF25		Weapon/munition combination parameter table for TACFIRE: 11 parameters for 36 weapon/munition combinations and two forces.
MUNDES(3,36,2) DF25		Munition description table for countermortar/counterbattery: three parameters for 36 weapon/munition combinations and two forces.

d. Logical Flow (Figure II-9-B-2):

(1) Blocks L50, L61, 1, L63, 2, and 3. The data file 29 data cards are read and card images copied to a temporary disk file (logical unit 30). Each card is read with a 20A4 format, assigned a sequence number, and printed. The card image is written to the temporary file with the last four characters replaced by the new sequence number. The last data card is expected to have a card ID of 9999 in card columns 73-76. The card image file is positioned to card image number one after the last card is copied.

(2) Block 4. NERR is the count of serious detectable data errors which will be returned to AFMLD; it is set to zero before the editing process begins.

(3) Block 5. Each group of cards with the same card ID (i.e., 2901, 2902, 2903, or 2904) must be preceded by a header card which is blank except for the card ID. The first card image must be a header card and establishes the card ID of the first group.

(4) Blocks L100 and 6. The card ID consists of four numeric characters: the first two are the file number (29), the last two are the card format number (01-04, 99). If the file number is 29, control goes to block L110. If the card format number is 99, it goes to block L199.

(5) Blocks 7 and 8. If the card ID is not 2901, 2902, 2903, 2904 or 9999, an error has occurred. An appropriate error message is printed and the error counter (NERR) is incremented.

(6) Block L110. This block represents a computed GO TO statement that branches to the appropriate code for reading the next card image according to the card format number.

(7) Block L120. Using format 2901, the following variables are filled from the next card image: ISIDE, IWMUNX, IWEOHX, IMEOHX, LAPERS(1-6), IFILE, IFORM, and ISEQ.

(8) Block 9. If IFORM is not equal to one, it indicates that the header card of the next card group has been read. Control branches to block L100.

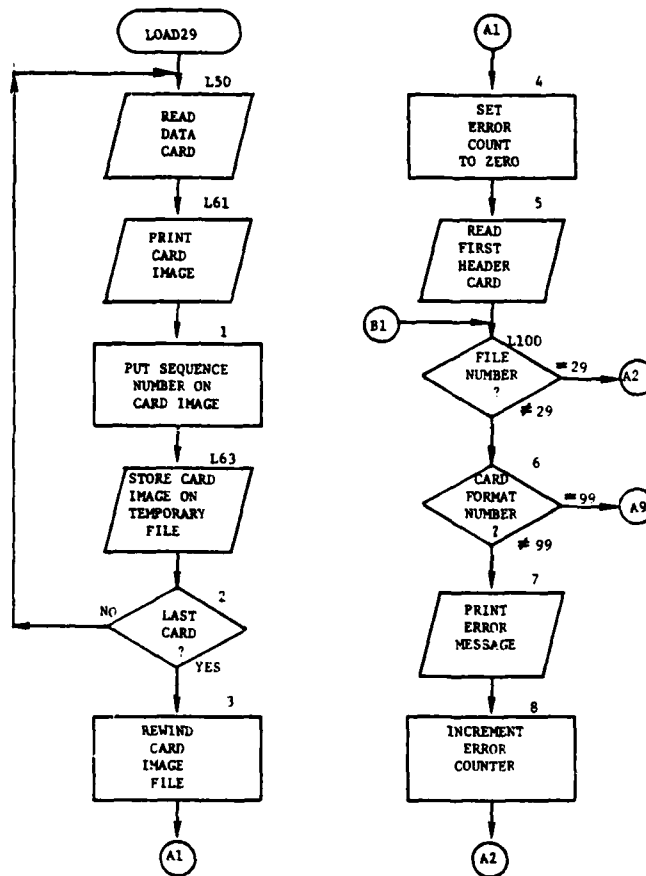


Figure II-9-B-2. Routine LOAD29 (Continued on Next Page)



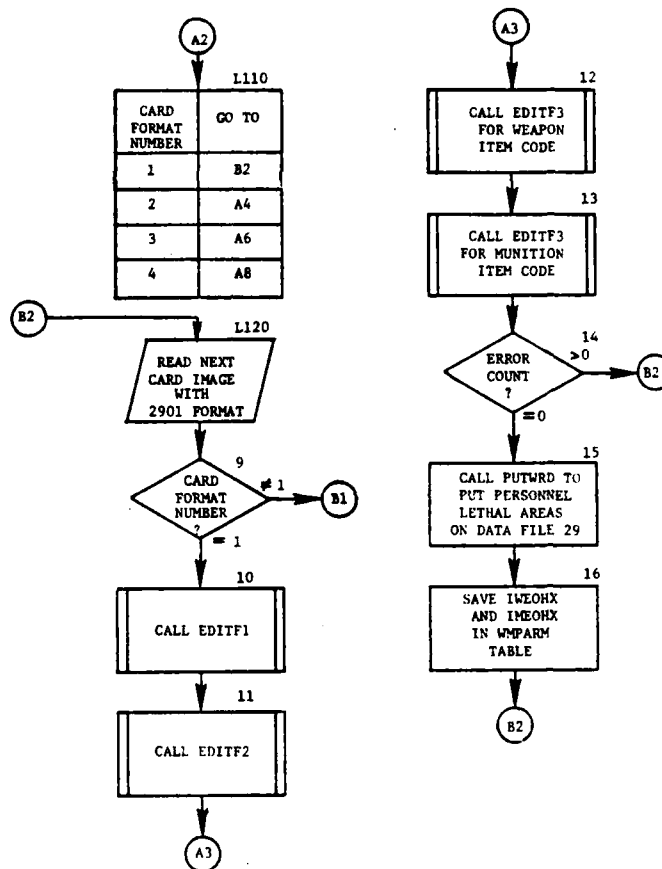


Figure II-9-B-2. Routine LOAD29 (Continued)

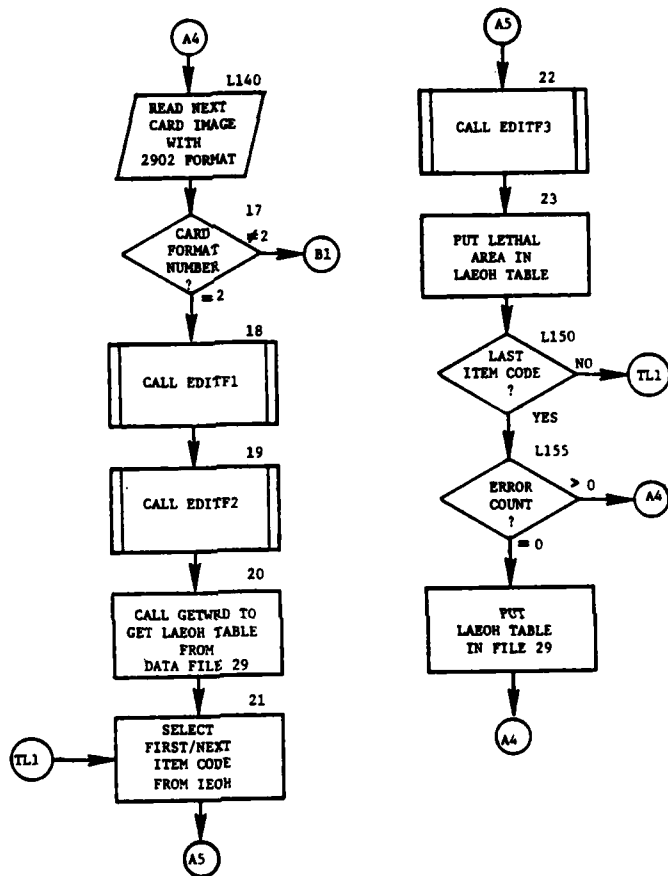


Figure II-9-B-2. Routine LOAD29 (Continued)

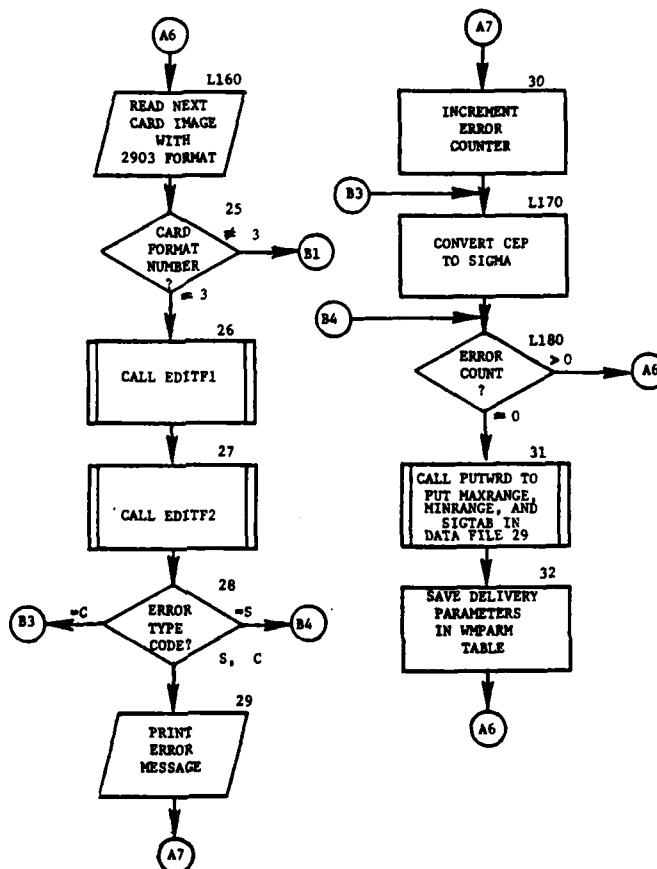


Figure II-9-B-2. Routine LOAD29 (Continued)

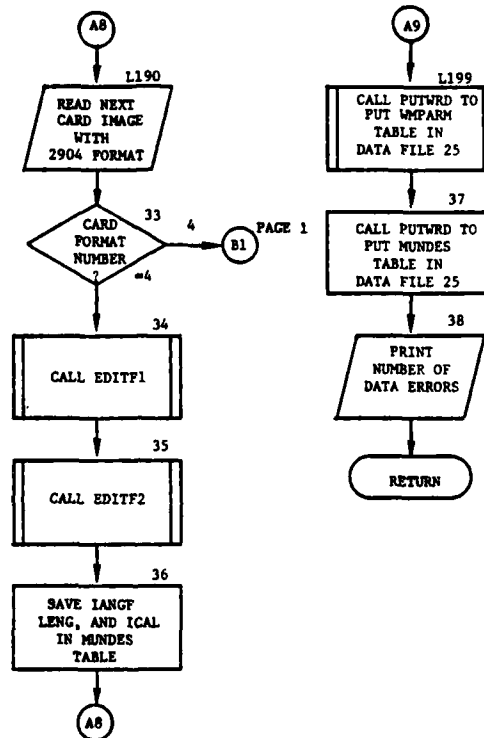


Figure II-9-B-2. Routine LOAD29 (Concluded)

(9) Blocks 10, 18, 26, and 34. Routine EDITF1 is called to edit the one-character force designator (ISIDE). If ISIDE is neither B nor R, then EDITF1 prints an error message and increments the error counter (NERR).

(10) Blocks 11, 19, 27, and 35. Routine EDITF2 is called to edit the weapon/munition combination index (IWMUNX). If IWMUNX is not between the values 1 and 36, EDITF2 prints an error message and increments the error counter.

(11) Blocks 12 and 13. EDITF3 is used to ensure that equipment item codes, in this case the weapon item code (IWEOHX) and the munition item code (IMEOHX), have values between 1 and 200. EDITF3 prints an error message and increments the error counter if the item code is not valid.

(12) Blocks 14, L155, and L180. If data errors have been found (NERR greater than zero), the loading of data file 29 is discontinued, although the card editing process continues.

(13) Block 15. IWEOHX, IMEOHX, and LAPERS(1-6) are loaded on data file 29 for this weapon/munition combination.

(14) Block 16. IWEOHX and IMEOHX are also placed in the weapon/munition parameter table (WMPARM) which is to be loaded on data file 25.

(15) Block L140. Using format 2902, the following variables are filled from the next card image: ISIDE, IWMUNX, IEOH(1-7), LAREA(1-7), IFILE, IFORM, and ISEQ.

(16) Block 17. If IFORM is not equal to two, it indicates that the header card of the next card group has been read and control goes to block L100.

(17) Blocks 20, 21, 22, 23, and L150. The table of lethal areas for equipment items (LAEOH) due to this weapon/munition combination is brought into core to have additional lethal areas filled. Seven item codes and lethal areas were read from the card image into IEOH and LAREA arrays. Each item code in IEOH is edited using the EDITF3 routine. The lethal areas in LAREA are put into the LAEOH array.

(18) Block 24. If no data errors have been found, the updated LAEOH array is returned to data file 29 and control returns to block L140 to read the next card image.

(19) Block L160. Using format 2903, the following variables are filled from the next card image: ISIDE, IWMUNX, MAXRNG, MINRNG, ETYPE, SIGTAB(1-6), DTV1, DTVMR, DTVSR, NRNDMR, EFFRAD, IFILE, IFROM, and ISEQ.

(20) Block 25. If IFORM is not equal to three, it indicates that the header card of the next card group has been read, and control goes to block L100.

(21) Blocks 28, 29, 30 and L170. The error type code (ETYPE) indicates whether the round dispersion errors (SIGTAB) are in the desired sigma form or the circular error probable (CEP) form; therefore, if ETYPE equals C, SIGTAB must be converted from CEPs to sigmas by multiplication of a conversion factor. Any other value of ETYPE is an error.

(22) Block 31. MAXRNG, MINRNG, and SIGTAB(1-6) are loaded onto data file 29 if no errors were detected.

(23) Block 32. DTV1, DTVMR, DTVSR, NRNDMR, MAXRNG, MINRNG, and EFFRAD are stored in the WMPARM table which will be loaded on data file 25 when completed.

(24) Block L190. Using format 2904, the following variables are filled from the next card image: ISIDE, IWMUNX, IANGF, LENG, ICAL, IFILE, IFORM, and ISEQ.

(25) Block 33. If IFORM is not equal to four, it indicates that the header card of the next card group has been read.

(26) Block 36. IANGF, LENG, and ICAL are saved in the MUNDES table which will be loaded on data file 25 when completed.

(27) Blocks L199 and 37. All data file 29 cards have been read and processed; therefore, the completed WMPARM and MUNDES arrays must be loaded into the appropriate locations of data file 25.

(28) Block 38. The number of data errors detected (NERR) is printed for the user.

#### 4. ROUTINE LOAD30:

a. Purpose. LOAD30 reads the personnel protection data required by the Area Fire Model and other models from cards and loads the data on DIVWAG data file 30.

##### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
KARD(20)	Card	Data card image buffer.
IFILE	Card	File number (30).
IFORM	Card	Card format number (1-4).
ISEQ	Card	Card image sequence number
ISIDE	Card	Force designator (B or R)
IACT(3)	Card	List of three activity indexes.

<u>Name</u>	<u>Source</u>	<u>Contents</u>
OPEND(3,7)	Card	Seven data items about unprotected (open) personnel for each of the three activity indexes given by IACT. The seven items are: percent standing unwarned, percent prone unwarned, percent in foxhole unwarned, percent standing warned, percent prone warned, percent in foxholes warned, and time to revert from warned to unwarned posture.
IAC TX	Card	Activity index (1-7).
IRANK(6)	Card	List of six protection priorities (orders of choice) between 1 and 100.
COVERD(6,3)	Card	Three data items about protected (covered) personnel for each of the six protection priorities. The three items are: the equipment item code, the number of personnel protected, and the number of casualties per item lost.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
NERR	Call	Number of serious data errors detected.
ISEQ	Print	Card image sequence number.
KARD(20)	Print	Data card image buffer.
UNPRO(7,7)	DF30	Table of unprotected personnel data; i.e., seven data items for each of seven activity indexes. The seven items are: percent standing unwarned, percent prone unwarned, percent in foxholes unwarned, percent standing warned, percent prone warned, and percent in foxholes warned, and time to revert from warned to unwarned postures.
EOHPRO(3,100)	DF30	Three data items (equipment item code, number of personnel per item, and number of casualties per item lost) for 100 protection priorities.

d. Logical Flow (Figure II-9-B-3):

(1) Blocks L50, L61, 1, L63, 2, and 3. The data file 30 cards are read and copied to a temporary disk file (logical unit 30). Each card is read with a 20A4 format, assigned a sequence number, and printed. The card

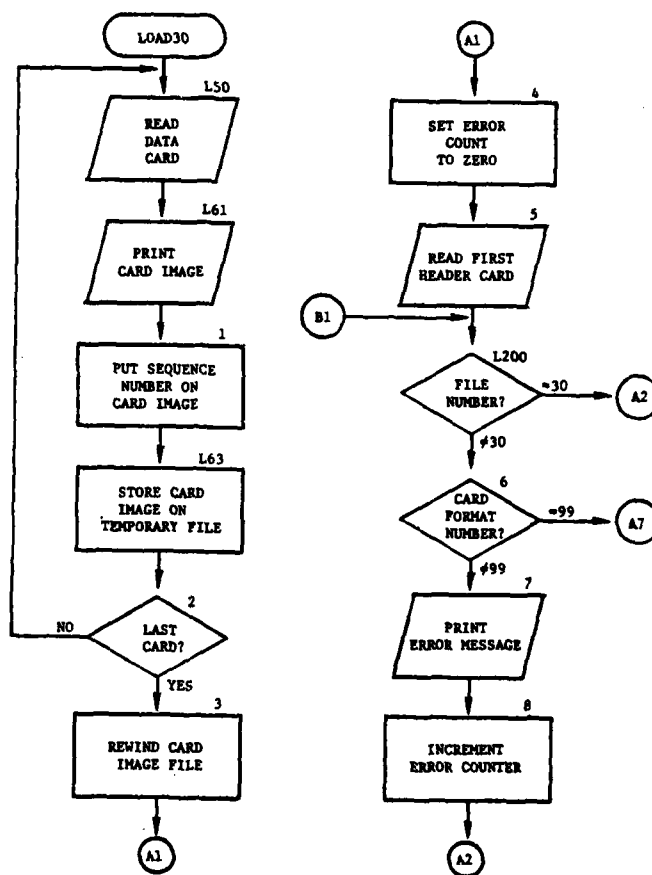


Figure II-9-B-3. Routine LOAD30 (Continued on Next Page)



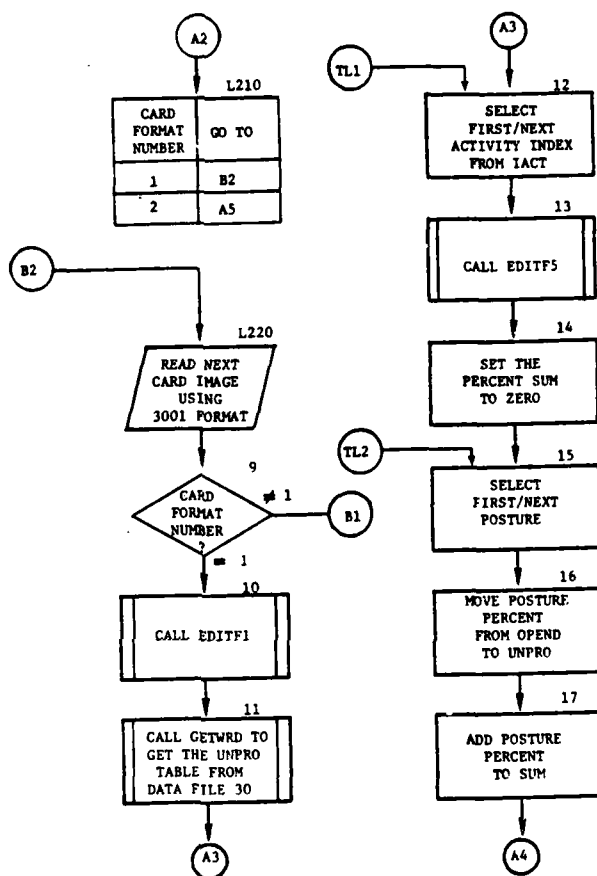


Figure II-9-B-3. Routine LOAD30 (Continued)

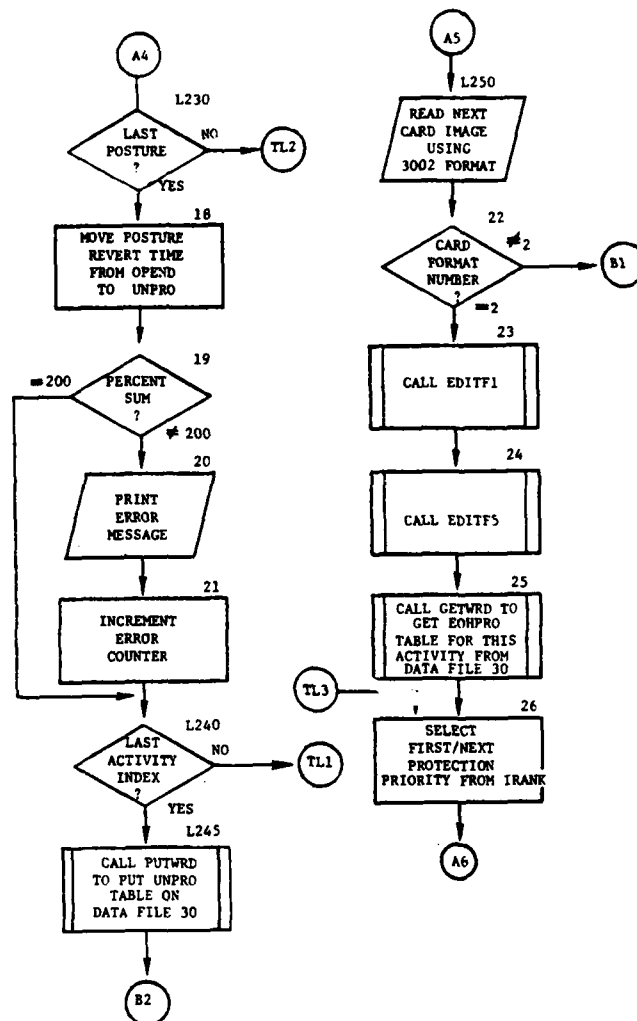


Figure II-9-B-3. Routine LOAD30 (Continued)

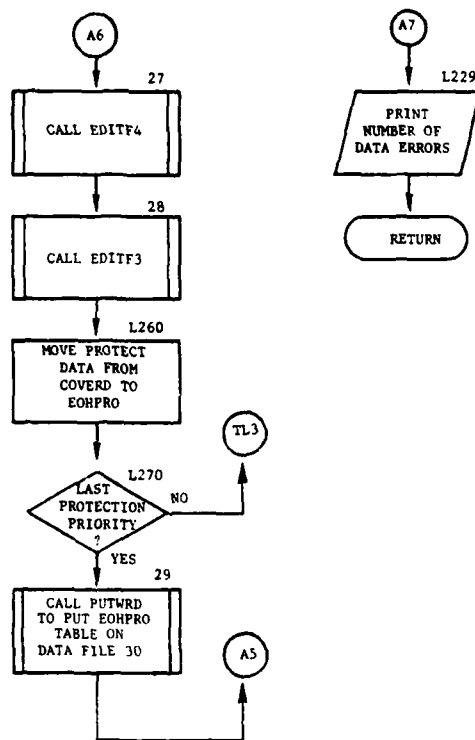


Figure II-9-B-3. Routine LOAD30 (Concluded)

image is written on the temporary file with the last four characters replaced by the new sequence number. The last data card has a card ID of 9999 in card columns 73-76. The card image file is positioned to the first card image after the last card is copied.

(2) Block 4. NERR is the number of serious detectable data errors which will be returned to AFMLD and it is set to zero before the editing process begins.

(3) Block 5. Each group of cards with the same card ID (i.e., 3001 or 3002) must be preceded by a header card which is blank except for the card ID.

(4) Block L200. The card ID consists of four numeric characters, the first two of which are the file number (30), the last two are the card format number (01, 02, 99). If the file number is 30, transfer control to block L210.

(5) Block 6. If the card format number is 99 transfer control to block L229.

(6) Blocks 7 and 8. If the card ID is not 3001, 3002, or 9999, an error has occurred. An appropriate error message is printed and the error counter (NERR) is incremented.

(7) Block L210. This block represents a computed GO TO statement which branches to appropriate code for reading the next card image according to the card format number. If the card format number is two, control is transferred to block L250. If the card format number is one, control goes to block L270.

(8) Block L220. Using format 3001, the following variables are filled from the next card image: ISIDE, IACT(7), OPEND(3,7), IFILE, IFORM, and ISEQ.

(9) Block 9. If IFORM is not equal to one, which indicates that the header card of the next card group has been read, control goes to block L200.

(10) Blocks 10 and 23. Routine EDITF1 is called to edit the one-character force designator (ISIDE). If ISIDE is neither B nor R, EDITF1 prints an error message and increments the error counter (NERR).

(11) Block 11. The UNPRO array is read from data file 30 to be updated with the new data.

(12) Block 12. The first/next activity index is selected from IACT(1-3).

(13) Block 13. The activity index from IACT is edited by routine EDITF5. If the index is not between one and seven, EDITF5 prints an error message and increments the error counter.

AD-A112 688

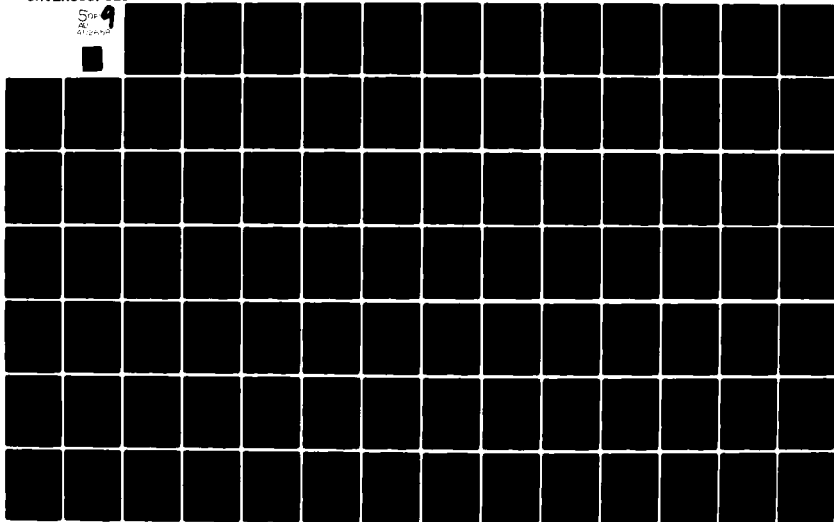
ARMY COMBINED ARMS COMBAT DEVELOPMENTS ACTIVITY FORT--ETC F/0 15/7  
DIVWAS MODEL DOCUMENTATION. VOLUME II. PROGRAMMER/ANALYST MANUAL--ETC(U)  
JUL 76

UNCLASSIFIED

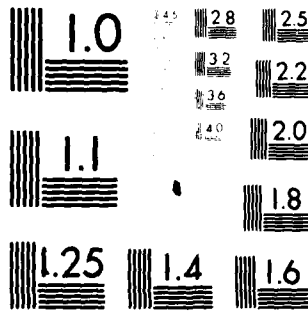
CACDA-TR-8-76-VOL-2-PT-1

NL

5010  
21  
01/01/76



11265



MICROCOPY RESOLUTION TEST CHART  
NATIONAL BUREAU OF STANDARDS-1963-A

(14) Block 14. The percent sum (IPCSUM) is used to verify that each unwarned and warned posture sums to 100 percent.

(15) Block 15. The first/next of six postures is selected: standing unwarned, prone unwarned, foxhole unwarned, standing warned, prone warned, and foxhole warned.

(16) Blocks 16 and 17. The percent of personnel in this posture is given in OPEND and is stored in UNPRO which will be returned to data file 30. The percent is also added to IPCSUM.

(17) Block L230. If the last posture has not been processed, control returns to block 15.

(18) Block 18. The seventh data item about unprotected personnel is the time to revert from warned to unwarned posture. This is taken from OPEND and stored in UNPRO.

(19) Blocks 19, 20 and 21. The percentages in the three unwarned postures total 100 percent, as do the percentages in the three warned postures; therefore, the total percent sum (IPCSUM) is 200; if it is not, an error message is printed and the error counter incremented.

(20) Blocks L240 and L245. If last activity index has not been processed control returns to block 12; otherwise the UNPRO table is placed into data file 30.

(21) Block L250. Using format 3002, the following variables are filled from the next card image; ISIDE, IACTX, IRANK(6), COVERD(6,3), IFILE, IFORM, and ISEQ.

(22) Block 22. If IFORM is not equal to two, it indicates that the header card of the next card group has been read, and control is transferred to block L200.

(23) Block 24. The activity index (IACTX) is edited by routine EDITF5. If the index is not between one and seven, EDITF5 prints an error message and increments the error counter.

(24) Block 25. EOHPRO(3,100) is the table of personnel protection data for a single activity. In this case it is taken from data file 30 according to IACTX.

(25) Block 26. The first/next protection priority (order of choice) is selected from the IRANK table.

(26) Block 27. Routine EDITF4 is called to edit the order-of-choice index. If the index is not between 1 and 100, EDITF4 prints an error message and increments the error counter.

(27) Block 28. Routine EDITF3 is called to edit the item code of the equipment item which is specified to provide personnel protection. If the item code is not between 1 and 200, EDITF3 prints an error message and increments the error counter (NERR).

(28) Block L260. The three protection data items for this protection priority are copied from the COVERD array to the EOHPRO array. The column index is identical to the order-of-choice index.

(29) Blocks L270 and 29. If the last protection priority has not been processed, control transfers to block 26; otherwise the EOHPRO table is returned to data file 30 and control is transferred to block L250.

(30) Block L299. After the last card image has been read, the number of data errors detected (NERR) is printed for the user.

5. ROUTINE DUMP29:

a. Purpose. DUMP29 prints the data which LOAD29 loaded onto the DIVWAG data files: (i.e., the contents of data file 29 and two selected portions of data file 25).

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
MUNDAT(216)	DF29	Munition data, data file 29 record.
IWEOHX	DF29	MUNDAT(1). Weapon item code.
IMEOHX	DF29	MUNDAT(2). Munition item code.
LAPERS(3,2)	DF29	MUNDAT(3). Lethal areas to personnel in three postures(standing, prone, foxhole) and two environments (unforested, forested).
LAEOH(200)	DF29	MUNDAT(9). Lethal areas for 200 equipment items.
MAXRNG	DF29	MUNDAT(209). Maximum range of munition.
MINRNG	DF29	MUNDAT(210). Minimum range of munition.
SIGTAB(6)	DF29	MUNDAT(211). Round dispersion errors at six range points between maximum range and minimum range.
WMPARM(36,11)	DF25	Weapon/munition combination parameter table for TACFIRE: eleven parameters for 36 weapon/munition combinations.



<u>Name</u>	<u>Source</u>	<u>Contents</u>
MUNDES(3,36)	DF25	Munition description table for countermortar/counterbattery: three data items for 36 weapon/munition combinations.

c. Output Variables. Same as input variables, destination being the printer.

d. Logical Flow (Figure II-9-B-4):

(1) Block 1. Each force is selected separately according to the force index (IFORCE); 1 for Blue, 2 for Red.

(2) Blocks 2 and 3. The weapon/munition combination parameter table (WMPARM) is taken from data file 25. It contains eleven words per weapon/munition combination, two of which are not used. This parameter table is printed.

(3) Blocks 4 and 5. The munition description table (MUNDES) is taken from data file 25. It contains three data items per weapon/munition combination. This table is printed.

(4) Block 6. Select the first/next of each of the 36 weapon/munition combinations by selecting the combination index (IMUN).

(5) Block 7. One record for each weapon/munition combination is brought into MUNDAT from data file 29.

(6) Block 8. The weapon/munition header consisting of the record number, the force name (RED or BLUE), the weapon/munition index (IMUN), the weapon item code (IWEOHX), and the munition item code (IMEOHX) are printed.

(7) Block 9. Six lethal areas against enemy personnel are listed for standing, prone, and foxhole postures, in both unforested and forested environments.

(8) Block L1000. An array (LAPRIN) is built containing nonzero lethal areas and the corresponding item codes so that only the nonzero lethal areas can be printed.

(9) Block L2000. The six round dispersal factors (SIGTAB) are printed with the six corresponding range points. The first range point is MAXRNG, the last is MINRNG, and the four intermediate ranges are calculated.

(10) Block L4000. If all of the weapon/munition combinations have not been processed, control is transferred to block 6.

(11) Block L5000. If Red force has not been processed, control transfers to block 1.

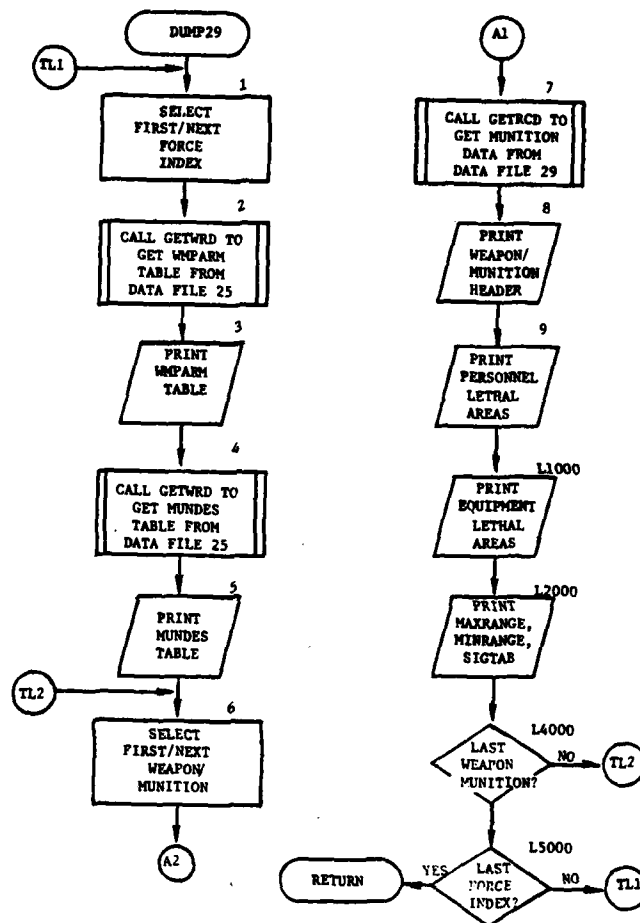


Figure II-9-B-4. Routine DUMP29

6. ROUTINE DUMP30:

a. Purpose. DUMP30 prints the data contained on data file 30, the personnel protection data.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
UNPRO(7,7)	DF30	Table of unprotected personnel data; i.e., seven data items for each of seven activity indexes. The seven items are: the percent of personnel in standing, prone, and foxhole postures, both unwarned and warned, and the time to revert from warned to unwarned postures.
EOHPRO(3,100)	DF30	Three data items (equipment item code, number of personnel per item, and casualties per item lost) for 100 protection priorities.

c. Output Variables. Same as input variables, destination being the printer.

d. Logical Flow (Figure II-9-B-5):

(1) Block 1. Select separately each of the two force indexes: 1 for Blue, 2 for Red.

(2) Block 2. The UNPRO table for the force is brought in from data file 30.

(3) Block 3. The first/next index of seven activity indexes is selected.

(4) Block 4. The activity header consisting of the record number, the force name (BLUE or RED), the activity index (IACTX), and the activity name taken from the IACT table are printed.

(5) Block 5. The personnel protection by equipment table (EOHPRO) is brought in from data file 30 for this activity index.

(6) Block 6. The percent of personnel in the three postures (standing, prone, foxhole) is printed from the UNPRO table for the two separate cases of unwarned and warned for this activity index.

(7) Block L2000. The order of choice, item code, number of personnel per item, and number of casualties per item lost are printed from the EOHPRO table for those items listed as protecting personnel.

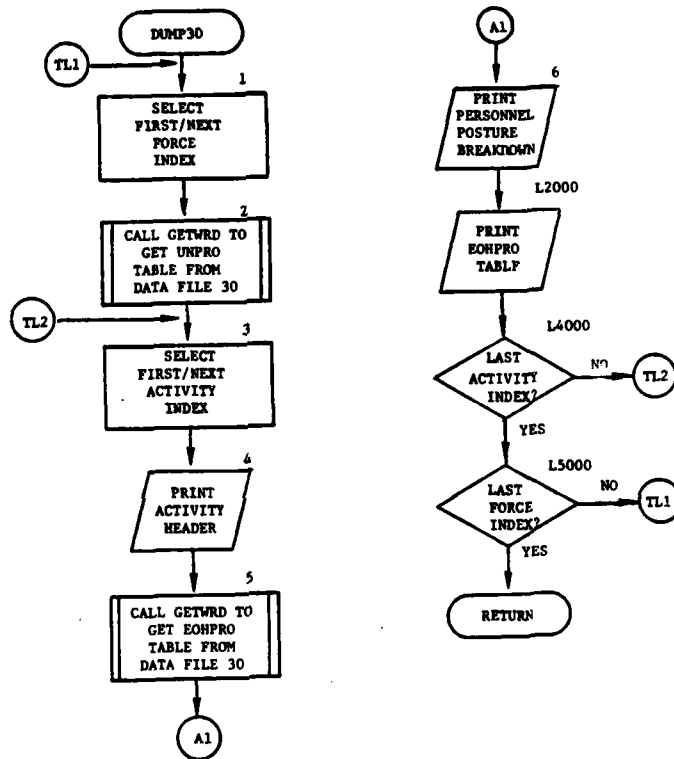


Figure II-9-B-5. Routine DUMP30

(8) Block L4000. If all of the activity indexes have not been processed, control is transferred to block 3.

(9) Block L5000. If the Red force has not been processed control transfers to block 1.

NOTES

II-9-B-26

## APPENDIX C

### AREA FIRE DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. Nine different printouts of constant data inputs are generated after the data have been read into the files. (It is coincident that there are nine output reports and nine card types for input.) These printouts are listed in Figure II-9-C-1.

Printout Title	Data Source
Data File 29 80-Column Card Image	Card Data Deck
Weapons and Munitions Characteristics and Performance	Data file 29
Weapons and Munitions Delivery Parameters	Data file 25
Weapons and Munitions Description	Data file 25
80-Column Card Image for Personnel Data	Card Data Deck
Personnel Data	Data file 30

Figure II-9-C-1. Area Fire Constant Data Printouts

2. DATA FILE 29 80-COLUMN CARD IMAGE. In Figure II-9-C-2 is an illustration of this format. At the top line, far left, are the characters 1B, indicating that the data are for the Blue force and represent index 1 in combining UTDs with common dimensions and distribution of personnel and equipment. At the far left is the card number, preceded by ID 2901. The data between the far right and far left columns are actually entered into the file.

3. WEAPONS AND MUNITIONS CHARACTERISTICS AND PERFORMANCE. This format is illustrated in Figure II-9-C-3. The upper third of the figure shows data originally recorded in card ID 2901; i.e., lethal area against Red personnel, both forested and unforested. The center of the figure shows lethal areas for Red force equipment data initially transcribed on card ID 2902. The lower third of the figure shows deflection and range errors for six different range distances beginning with maximum at the left and proceeding to minimum at the right.

4. WEAPONS AND MUNITIONS DELIVERY PARAMETERS. This format is illustrated in Figure II-9-C-4. The weapon item code and the munition item code are read from card ID 2901 and the remaining data are read from card ID 2903. The columnar headings designate the following data. The data are stored in data file 25.

FILE-29 DATA CARD IMAGES

18	1	25	27	450	233	5	275	104	2	2901	CARD 1
18	2	26	27	450	233	5	275	104	2	2901	CARD 2
18	3	29	30	867	723	152	452	309	42	2901	CARD 3
18	4	50	51	867	723	152	452	309	42	2901	CARD 4
18	5	52	53	9911	7653	113	5058	3422	51	2901	CARD 5
18	6	52	54	6239	3129	85	3591	1873	57	2901	CARD 6
18	7	55	56	9401	7382	113	4949	3374	51	2901	CARD 7
18	8	55	56	6067	3093	85	3538	1859	57	2901	CARD 8
18	9	56	57	11268	6308	188	7130	3928	125	2901	CARD 9
18	10	58	59	1355	1004	124	577	248	35	2901	CARD 10
18	11	60	61	212224	117197	2375	92532	5195	1049	2901	CARD 11
18	12	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 12
18	13	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 13
18	14	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 14
18	15	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 15
18	16	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 16
18	17	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 17
18	18	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 18
18	19	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 19
18	20	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 20
18	21	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 21
18	22	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 22
18	23	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 23
18	24	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 24
18	25	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 25
18	26	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 26
18	27	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 27
18	28	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 28
18	29	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 29
18	30	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 30
18	31	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 31
18	32	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 32
18	33	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 33
18	34	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 34
18	35	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 35
18	36	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 36
18	37	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 37
18	38	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 38
18	39	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 39
18	40	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 40
18	41	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 41
18	42	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 42
18	43	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 43
18	44	61	62	212224	117197	2375	92532	5195	1049	2902	CARD 44

II-9-C-2

Figure II-9-C-2. Data File 29 80-Column Card Image



LETHAL AREA AGAINST REQ	PERSONNEL, NO	FOREST %	STANDING	PRONE	FOXHOLE
			450	200	5
					150. METERS
LETHAL AREA AGAINST REQ	PERSONNEL, NO	FOREST %	STANDING	PRONE	FOXHOLE
			275	104	2
					150. METERS

LFTAL AREAS AGAINST RED EQUIPMENT		150. MEYESS	
ITEM CODE	LFTAL AREA	ITEM CODE	LFTAL AREA
21	2	22	9
23	9	32	207
36	5	38	18
54	9	56	9
63	6	70	18
73	36	73	71
85	36	75	61
104	104	87	134
110	63	94	104
114	450	111	450
119	450	115	450
122	450	119	450
123	63	123	63
126	63	135	135
140	18	141	141
144	9	145	145
148	63	149	36
151	5	162	5
165	18	166	18
169	18	170	18
173	18	174	207
175	9	177	18
186	186	187	18
		23	9
		33	200
		50	9
		58	104
		71	18
		76	450
		88	63
		112	450
		113	450
		117	450
		121	450
		125	63
		139	18
		143	104
		147	5
		164	9
		165	18
		168	207
		172	18
		176	176
		178	5
		27	5
		35	5
		52	9
		60	104
		72	18
		78	450
		88	63
		113	450
		117	450
		121	450
		125	63
		139	18
		143	104
		147	5
		164	9
		165	18
		168	207
		172	18
		176	176
		178	5

OFFLECTION AND RANGE ERROR (SIGMA) AS FUNCTION OF RANGE

QWAGE (WETQPS) %	MINRANGE 4500	MINRANGE 1010	MINRANGE 940	MINRANGE 50
SIGMA (WETQPS) %	18	16	12	9

**Figure II-9-C-3. Weapons and Munitions Characteristics and Performance**

ALUF WEAPON/MUNITION DELIVERY PARAMETERS (FILE-25)

WEAPON/MUNITION INDEX	WEAPON ITEM CODE	MUNITION ITEM CODE	SECONDS TO FIRE	SECONDS PER ROUND	SECONDS PER ROUND SUSTAINED RATE	ROUND LIMIT MAXIMUM RATE	MAXIMUM RANGE	MINIMUM RANGE	EFFECTS RADIUS
1	25	27	120	5	20	30	4500	50	150
2	26	27	120	5	20	30	4500	50	150
3	29	30	120	8	30	20	5500	840	175
4	50	51	120	6	20	30	11500	1000	175
5	52	53	120	15	60	12	18000	2500	360
6	52	54	120	15	60	12	18000	2500	360
7	55	65	120	15	60	12	14600	2000	360
8	55	66	120	15	60	12	14600	2000	360
9	56	57	120	45	120	6	22000	4200	470
10	58	59	120	45	120	4	32700	4200	510
11	60	61	1500	1200	1200	1	34000	5000	450
12	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	0	0
34	0	0	0	0	0	0	0	0	0
35	0	0	0	0	0	0	0	0	0
36	0	0	0	0	0	0	0	0	0

II-9-C-4

Figure II-9-C-4. Weapons and Munitions Delivery Parameters

5. WEAPONS AND MUNITIONS DESCRIPTION. This format is illustrated in Figure II-9-C-5. The information is read from card ID 2904 and stored in data file 25. The mean angle of fire is printed in decidegrees and represents the angle between the horizontal and the direction of fire. The projectile length and caliber are printed in millimeters in the third and fourth columns, respectively.

6. EIGHTY-COLUMN CARD IMAGE, PERSONNEL DATA. This format is shown in Figure II-9-C-6. It lists the card image of data for loading data file 30 with personnel data.

7. PERSONNEL DATA. The personnel data, the postures that they may assume in combat, and the type of equipment available to afford protection are shown on this printout, Figure II-9-C-7. The table in the upper part of the figure has data originally transcribed in card ID 3001. The lower half of the figure has data in card ID 3002.

BLUFWEAPON/MUNITION DESCRIPTION (FILE-25)

WEAPON/MUNITION INDEX	MEAN ANGLE OF FIRE	LENGTH MM	CALIBER MM
1	500.	528	81
2	500.	528	81
3	500.	655	107
4	190.	831	105
5	180.	700	155
6	180.	700	155
7	290.	700	155
8	290.	700	155
9	270.	892	203
10	180.	1732	175
11	140.	13563	762
12	0.	0	0
13	0.	0	0
14	0.	0	0
15	0.	0	0
16	0.	0	0
17	0.	0	0
18	0.	0	0
19	0.	0	0
20	0.	0	0
21	0.	0	0
22	0.	0	0
23	0.	0	0
24	0.	0	0
25	0.	0	0
26	0.	0	0
27	0.	0	0
28	0.	0	0
29	0.	0	0
30	0.	0	0
31	0.	0	0
32	0.	0	0
33	0.	0	0
34	0.	0	0
35	0.	0	0
36	0.	0	0

Figure II-9-C-5. Weapons and Munitions Description

II-9-C-6

FILE-36 DATA CARD IMAGES

CARD	1	3001	1	CARD	2	3001	2	CARD	3	3001	3	CARD	4	3001	4	CARD	5	3001	5	CARD	6	3001	6	CARD	7	3001	7	CARD	8	3001	8	CARD	9	3001	9	CARD	10	3001	10	CARD	11	3001	11	CARD	12	3001	12	CARD	13	3001	13	CARD	14	3001	14	CARD	15	3001	15	CARD	16	3001	16	CARD	17	3001	17	CARD	18	3001	18	CARD	19	3001	19	CARD	20	3001	20	CARD	21	3001	21	CARD	22	3001	22	CARD	23	3001	23	CARD	24	3001	24	CARD	25	3001	25	CARD	26	3001	26	CARD	27	3001	27	CARD	28	3001	28	CARD	29	3001	29	CARD	30	3001	30	CARD	31	3001	31	CARD	32	3001	32	CARD	33	3001	33	CARD	34	3001	34	CARD	35	3001	35	CARD	36	3001	36	CARD	37	3001	37	CARD	38	3001	38	CARD	39	3001	39	CARD	40	3001	40	CARD	41	3001	41	CARD	42	3001	42	CARD	43	3001	43	CARD	44	3001	44	CARD	45	3001	45	CARD	46	3001	46	CARD	47	3001	47	CARD	48	3001	48	CARD	49	3001	49	CARD	50	3001	50	CARD	51	3001	51	CARD	52	3001	52	CARD	53	3001	53	CARD
CARD	1	3001	1	CARD	2	3001	2	CARD	3	3001	3	CARD	4	3001	4	CARD	5	3001	5	CARD	6	3001	6	CARD	7	3001	7	CARD	8	3001	8	CARD	9	3001	9	CARD	10	3001	10	CARD	11	3001	11	CARD	12	3001	12	CARD	13	3001	13	CARD	14	3001	14	CARD	15	3001	15	CARD	16	3001	16	CARD	17	3001	17	CARD	18	3001	18	CARD	19	3001	19	CARD	20	3001	20	CARD	21	3001	21	CARD	22	3001	22	CARD	23	3001	23	CARD	24	3001	24	CARD	25	3001	25	CARD	26	3001	26	CARD	27	3001	27	CARD	28	3001	28	CARD	29	3001	29	CARD	30	3001	30	CARD	31	3001	31	CARD	32	3001	32	CARD	33	3001	33	CARD	34	3001	34	CARD	35	3001	35	CARD	36	3001	36	CARD	37	3001	37	CARD	38	3001	38	CARD	39	3001	39	CARD	40	3001	40	CARD	41	3001	41	CARD	42	3001	42	CARD	43	3001	43	CARD	44	3001	44	CARD	45	3001	45	CARD	46	3001	46	CARD	47	3001	47	CARD	48	3001	48	CARD	49	3001	49	CARD	50	3001	50	CARD	51	3001	51	CARD	52	3001	52	CARD	53	3001	53	CARD
CARD	1	3001	1	CARD	2	3001	2	CARD	3	3001	3	CARD	4	3001	4	CARD	5	3001	5	CARD	6	3001	6	CARD	7	3001	7	CARD	8	3001	8	CARD	9	3001	9	CARD	10	3001	10	CARD	11	3001	11	CARD	12	3001	12	CARD	13	3001	13	CARD	14	3001	14	CARD	15	3001	15	CARD	16	3001	16	CARD	17	3001	17	CARD	18	3001	18	CARD	19	3001	19	CARD	20	3001	20	CARD	21	3001	21	CARD	22	3001	22	CARD	23	3001	23	CARD	24	3001	24	CARD	25	3001	25	CARD	26	3001	26	CARD	27	3001	27	CARD	28	3001	28	CARD	29	3001	29	CARD	30	3001	30	CARD	31	3001	31	CARD	32	3001	32	CARD	33	3001	33	CARD	34	3001	34	CARD	35	3001	35	CARD	36	3001	36	CARD	37	3001	37	CARD	38	3001	38	CARD	39	3001	39	CARD	40	3001	40	CARD	41	3001	41	CARD	42	3001	42	CARD	43	3001	43	CARD	44	3001	44	CARD	45	3001	45	CARD	46	3001	46	CARD	47	3001	47	CARD	48	3001	48	CARD	49	3001	49	CARD	50	3001	50	CARD	51	3001	51	CARD	52	3001	52	CARD	53	3001	53	CARD
CARD	1	3001	1	CARD	2	3001	2	CARD	3	3001	3	CARD	4	3001	4	CARD	5	3001	5	CARD	6	3001	6	CARD	7	3001	7	CARD	8	3001	8	CARD	9	3001	9	CARD	10	3001	10	CARD	11	3001	11	CARD	12	3001	12	CARD	13	3001	13	CARD	14	3001	14	CARD	15	3001	15	CARD	16	3001	16	CARD	17	3001	17	CARD	18	3001	18	CARD	19	3001	19	CARD	20	3001	20	CARD	21	3001	21	CARD	22	3001	22	CARD	23	3001	23	CARD	24	3001	24	CARD	25	3001	25	CARD	26	3001	26	CARD	27	3001	27	CARD	28	3001	28	CARD	29	3001	29	CARD	30	3001	30	CARD	31	3001	31	CARD	32	3001	32	CARD	33	3001	33	CARD	34	3001	34	CARD	35	3001	35	CARD	36	3001	36	CARD	37	3001	37	CARD	38	3001	38	CARD	39	3001	39	CARD	40	3001	40	CARD	41	3001	41	CARD	42	3001	42	CARD	43	3001	43	CARD	44	3001	44	CARD	45	3001	45	CARD	46	3001	46	CARD	47	3001	47	CARD	48	3001	48	CARD	49	3001	49	CARD	50	3001	50	CARD	51	3001	51	CARD	52	3001	52	CARD	53	3001	53	CARD
CARD	1	3001	1	CARD	2	3001	2	CARD	3	3001	3	CARD	4	3001	4	CARD	5	3001	5	CARD	6	3001	6	CARD	7	3001	7	CARD	8	3001	8	CARD	9	3001	9	CARD	10	3001	10	CARD	11	3001	11	CARD	12	3001	12	CARD	13	3001	13	CARD	14	3001	14	CARD	15	3001	15	CARD	16	3001	16	CARD	17	3001	17	CARD	18	3001	18	CARD	19	3001	19	CARD	20	3001	20	CARD	21	3001	21	CARD	22	3001	22	CARD	23	3001	23	CARD	24	3001	24	CARD	25	3001	25	CARD	26	3001	26	CARD	27	3001	27	CARD	28	3001	28	CARD	29	3001	29	CARD	30	3001	30	CARD	31	3001	31	CARD	32	3001	32	CARD	33	3001	33	CARD	34	3001	34	CARD	35	3001	35	CARD	36	3001	36	CARD	37	3001	37	CARD	38	3001	38	CARD	39	3001	39	CARD	40	3001	40	CARD	41	3001	41	CARD	42	3001	42	CARD	43	3001	43	CARD	44	3001	44	CARD	45	3001	45	CARD	46	3001	46	CARD	47	3001	47	CARD	48	3001	48	CARD	49	3001	49	CARD	50	3001	50	CARD	51	3001	51	CARD	52	3001	52	CARD	53	3001	53	CARD
CARD	1	3001	1	CARD	2	3001	2	CARD	3	3001	3	CARD	4	3001	4	CARD	5	3001	5	CARD	6	3001	6	CARD	7	3001	7	CARD	8	3001	8	CARD	9	3001	9	CARD	10	3001	10	CARD	11	3001	11	CARD	12	3001	12	CARD	13	3001	13	CARD	14	3001	14	CARD	15	3001	15	CARD	16	3001	16	CARD	17	3001	17	CARD	18	3001	18	CARD	19	3001	19	CARD	20	3001	20	CARD	21	3001	21	CARD	22	3001	22	CARD	23	3001	23	CARD	24	3001	24	CARD	25	3001	25	CARD	26	3001	26	CARD	27	3001	27	CARD	28	3001	28	CARD	29	3001	29	CARD	30	3001	30	CARD	31	3001	31	CARD	32	3001	32	CARD	33	3001	33	CARD	34	3001	34	CARD	35	3001	35	CARD	36	3001	36	CARD	37	3001	37	CARD	38	3001	38	CARD	39	3001	39	CARD	40	3001	40	CARD	41	3001	41	CARD	42	3001	42	CARD	43	3001	43	CARD	44	3001	44	CARD	45	3001	45	CARD	46	3001	46	CARD	47	3001	47	CARD	48	3001	48	CARD	49	3001	49	CARD	50	3001	50	CARD	51	3001	51	CARD	52	3001	52	CARD	53	3001	53	CARD
CARD	1	3001	1	CARD	2	3001	2	CARD	3	3001	3	CARD	4	3001	4	CARD	5	3001	5	CARD	6	3001	6	CARD	7	3001	7	CARD	8	3001	8	CARD	9	3001	9	CARD	10	3001	10	CARD	11	3001	11	CARD	12	3001	12	CARD	13	3001	13	CARD	14	3001	14	CARD	15	3001	15	CARD	16	3001	16	CARD	17	3001	17	CARD	18	3001	18	CARD	19	3001	19	CARD	20	3001	20	CARD	21	3001	21	CARD	22	3001	22	CARD	23	3001	23	CARD	24	3001	24	CARD	25	3001	25	CARD	26	3001	26	CARD	27	3001	27	CARD	28	3001	28	CARD	29	3001	29	CARD	30	3001	30	CARD	31	3001	31	CARD	32	3001	32	CARD	33	3001	33	CARD	34	3001	34	CARD	35	3001	35	CARD	36	3001	36	CARD	37	3001	37	CARD	38	3001	38	CARD	39	3001	39	CARD	40	3001	40	CARD	41	3001	41	CARD	42	3001	42	CARD	43	3001	43	CARD	44	3001	44	CARD	45	3001	45	CARD	46	3001	46	CARD	47	3001	47	CARD	48	3001	48	CARD	49	3001	49	CARD	50	3001	50	CARD	51	3001	51	CARD	52	3001	52	CARD	53	3001	53	CARD
CARD	1	3001	1	CARD	2	3001	2	CARD	3	3001	3	CARD	4	3001	4	CARD	5	3001	5	CARD	6	3001	6	CARD	7	3001	7	CARD	8	3001	8	CARD	9	3001	9	CARD	10	3001	10	CARD	11	3001	11	CARD	12	3001	12	CARD	13	3001	13	CARD	14	3001	14	CARD	15	3001	15	CARD	16	3001	16	CARD	17	3001	17	CARD	18	3001	18	CARD	19	3001	19	CARD	20	3001	20	CARD	21	3001	21	CARD	22	3001	22	CARD	23	3001	23	CARD	24	3001	24	CARD	25	3001	25	CARD	26	3001	26	CARD	27	3001	27	CARD	28	3001	28	CARD	29	3001	29	CARD	30	3001	30	CARD	31	3001	31	CARD	32	3001	32	CARD	33	3001	33	CARD	34	3001	34	CARD	35	3001	35	CARD	36	3001	36	CARD	37	3001	37	CARD	38	3001	38	CARD	39	3001	39	CARD	40	3001	40	CARD	41	3001	41	CARD	42	3001	42	CARD	43	3001	43	CARD	44	3001	44	CARD	45	3001	45	CARD	46	3001	46	CARD	47	3001	47	CARD	48	3001	48	CARD	49	3001	49	CARD	50	3001	50	CARD	51	3001	51	CARD	52	3001	52	CARD	53	3001	53	CARD
CARD	1	3001	1	CARD	2	3001	2	CARD	3	3001	3	CARD	4	3001	4	CARD	5	3001	5	CARD	6																																																																																																																																																																																															

Figure II-9-C-6. Data File 30 80-Column Card Image

RECORD 1

BLUE PERSONNEL PROTECTION DATA

ACTIVITY INDEX 1 (STAY 1)

PERSONNEL NOT PROTECTED BY EQUIPMENT

POSTURE BREAKDOWN, UNWARNED %	STANDING	PRONE	FOXHOLE
	30	60	10
POSTURE BREAKDOWN, WARNED %	STANDING	PRONE	FOXHOLE
	3	80	17
TIME TO RETURN TO UNWARNED POSTURE %	5 (MINUTES)		

PROTECTION OF PERSONNEL BY EQUIPMENT

PROTECTION PRIORITY	ITEM CODE	PERSONNEL PER ITEM	CASUALTIES PER LOSS
1	37	4	4
2	174	4	4
3	140	2	2
4	41	4	4
5	185	11	11
6	29	5	5
7	35	4	4
8	25	5	5
9	186	3	3
10	71	5	5
11	75	4	4
12	188	5	5
13	189	2	2
14	52	10	10
15	18	3	3

II-9-C-8

Figure II-9-C-7. Personnel Data

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR AREA FIRE

(AVAILABLE UNDER SEPARATE COVER)

II-9-D-1

NOTES

II-9-D-2



## CHAPTER 10

### TACFIRE DATA INPUT PROGRAMS

1. INTRODUCTION. The TACFIRE data input programs create and load the data file required by the TACFIRE Model. These data consist of target priority and method of attack tables.

#### 2. ROUTINES:

a. General. The routines described in the following paragraphs comprise the TACFIRE data input programs.

b. Routine TACLD. This routine controls the other routines of the TACFIRE data input. It also creates and initializes the required data file. After that has been completed, it calls utility routines LOAD25 and DUMP25.

c. Routine LOAD25. This routine is responsible for the reading and editing of the data cards. It also stores the data extracted from the cards in the appropriate position on the file.

d. Routine OSORT. This routine is called by LOAD25 for the purpose of ranking a list of values in ascending order.

e. Routine DUMP25. This routine provides a listing of the file contents in tabular form.

3. FILES. Data file 25 is loaded by the TACFIRE data input programs.

NOTES

II-10-2

## APPENDIX A

### TACFIRE DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for tactical fire direction systems are documented in Appendix A, TACFIRE Model Input Requirements, to Chapter 9 of the Period Processor (Section IV).

NOTES

II-10-A-2

## APPENDIX B

### TACFIRE DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. The input data of the TACFIRE Model are loaded by the routines described in this appendix. TACLD initializes the data file and calls routine LOAD25 to read data cards and load the data, and routine DUMP25 to provide a formatted listing of the data. LOAD25 is supported by a utility sort routine, OSORT.

#### 2. ROUTINE TACLD:

a. Purpose. Program TACLD is the driver for loading constant data target priority and method of attack tables required by the DIVWAG TACFIRE Model. These data are loaded onto data file 25.

b. Input Variables. Input variables consist of TACFIRE data entered on cards 2501, 2502, and 2503. (Refer to LOAD25 routine.)

c. Output Variables. TACFIRE data stored on data file 25. (Refer to DUMP25 routine.)

#### d. Logical Flow (Figure II-10-B-1):

(1) Block 1. A call is made to routine GETFLE to retrieve the file name table.

(2) Block 2. A call is made to the routine CREATE to create the TACFIRE record on data file 25 with one record of 27092 words.

(3) Block L90. Initialize priority area of data file 25.

(4) Block L100. Initialize method of attack area of data file 25.

(5) Block 3. Call routine LOAD25 to read the data cards and load data file 25.

(6) Block 4. If no errors were detected in LOAD25, a call is made to routine DUMP25 to list data file 25.

(7) Block 5. The file name table is listed and the routine is terminated.

#### 3. ROUTINE LOAD25:

a. Purpose. Routine LOAD25 loads the TACFIRE Model's target priority and method of attack tables onto data file 25.

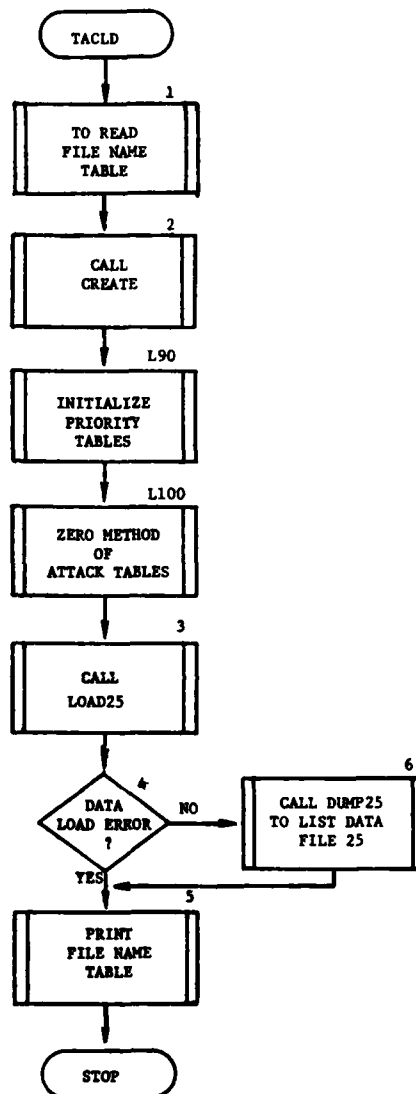


Figure II-10-B-1. Routine TACLD

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
KARD(20)	Card	TACFIRE Model's data file 25 entries on card formats 2501, 2502, and 2503.
IRNGP(4,5,77)	Card	TACFIRE Model's target priority data for each of four possible target sizes and 77 activity-type combinations.
MTHATK(20,5,77)	Card	TACFIRE Model's method of attack table array. A table of 20 choices exists for each of the 77 possible activity-type target combinations.
NERR	Call	Error counter index.
KFORCE	Card	Force indicator on card being processed.
JFORCE	Card	Force indicator of previous card processed.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
IRNGP(4,5,77)	DF25	Target priority tables.
MTHATK(20,5,77)	DF25	Method of attack tables.

d. Logical Flow (Figure II-10-B-2):

(1) Blocks L50, 1, 2, and 3. The logic of LOAD25 begins by reading each data load card, printing the card image, and storing the card image on a temporary scratch file.

(2) Block L70, 6, 7, 8, L76, and L80. The scratch file containing the TACFIRE card images is read and edited. Each card image is read. If the card image is a 2502 type all nonnumeric data are deleted. All card images are then rewritten to another scratch file.

(3) Block 4. Initialization of the data arrays in common is accomplished by setting the entries of the target priority table array, IRNGP, to 100. These values are used in the model to indicate that no priorities have been assigned to the target in the data load. Valid target combinations are given priorities in the load data with values 1-99. The method of attack array, MTHATK, is zeroed in the initialization process.

(4) Blocks 5 and 9. The first card in the deck is read and checked. If it is not a 2501 group card with no data entries, transfer control to block L210.

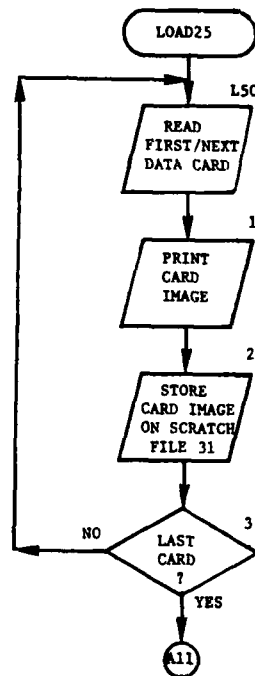


Figure II-10-B-2. Routine LOAD25 (Continued on Next Page)



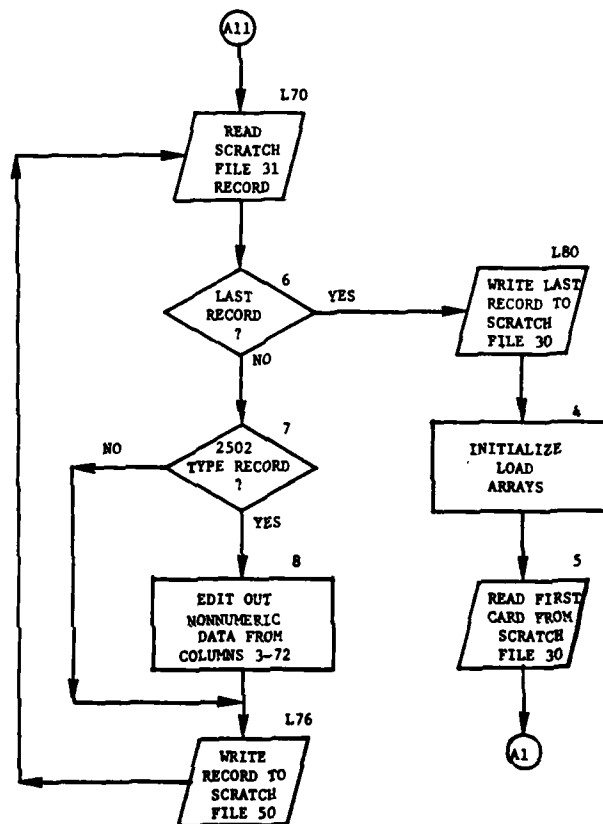


Figure II-10-B-2. Routine LOAD25 (Continued)

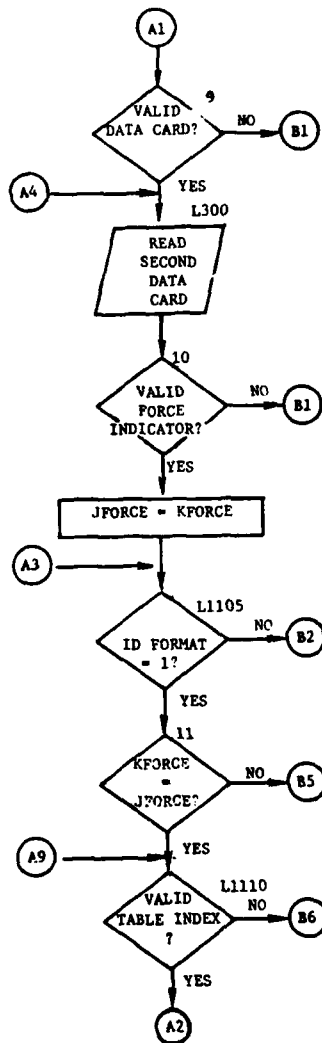


Figure II-10-B-2. Routine LOAD25 (Continued)

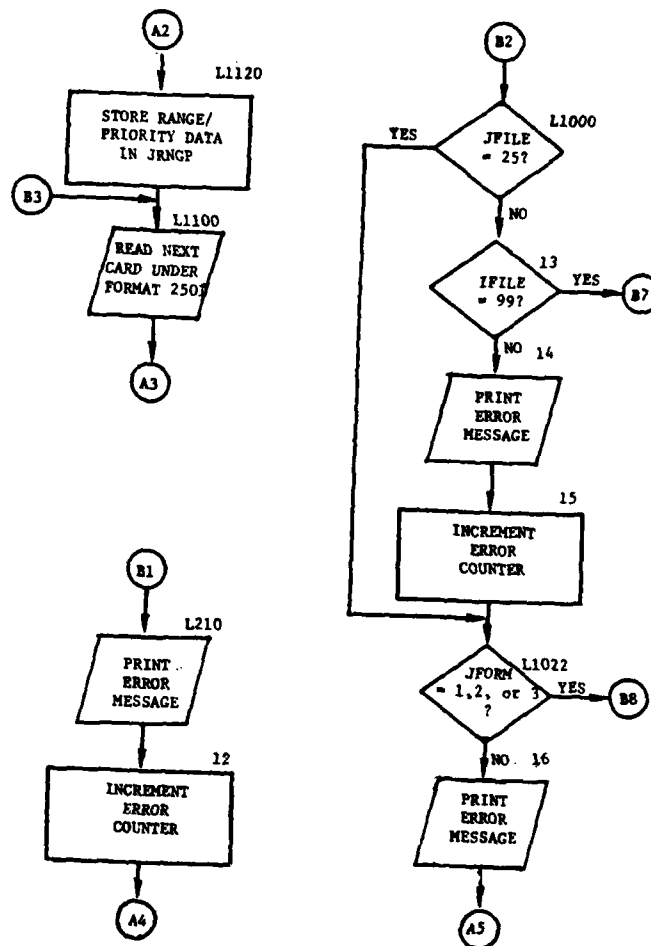


Figure II-10-B-2. Routine LOAD25 (Continued)

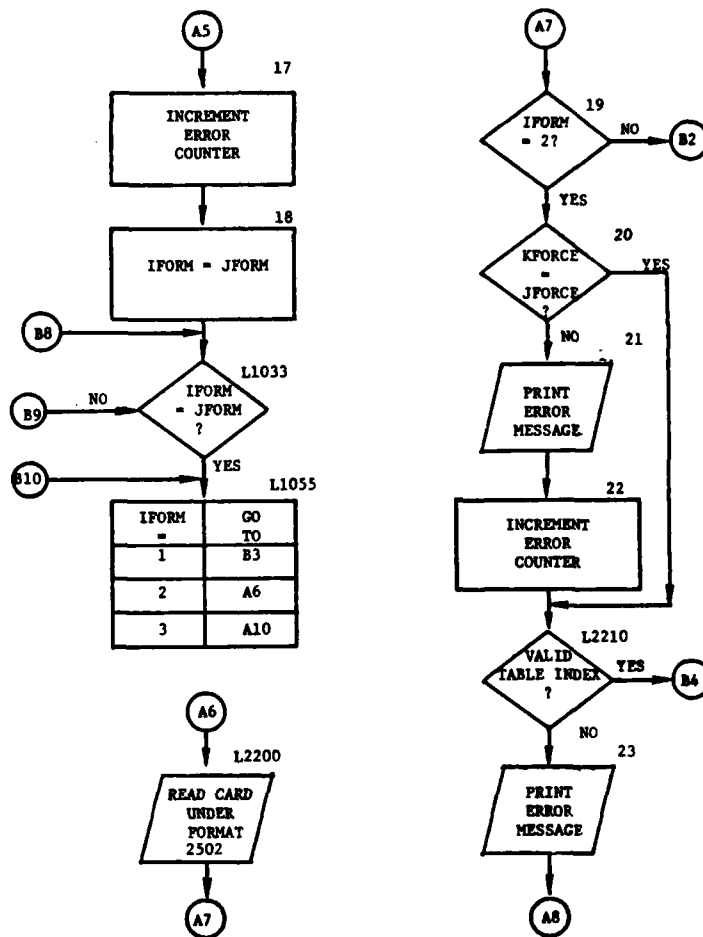


Figure II-10-B-2. Routine LOAD25 (Continued)

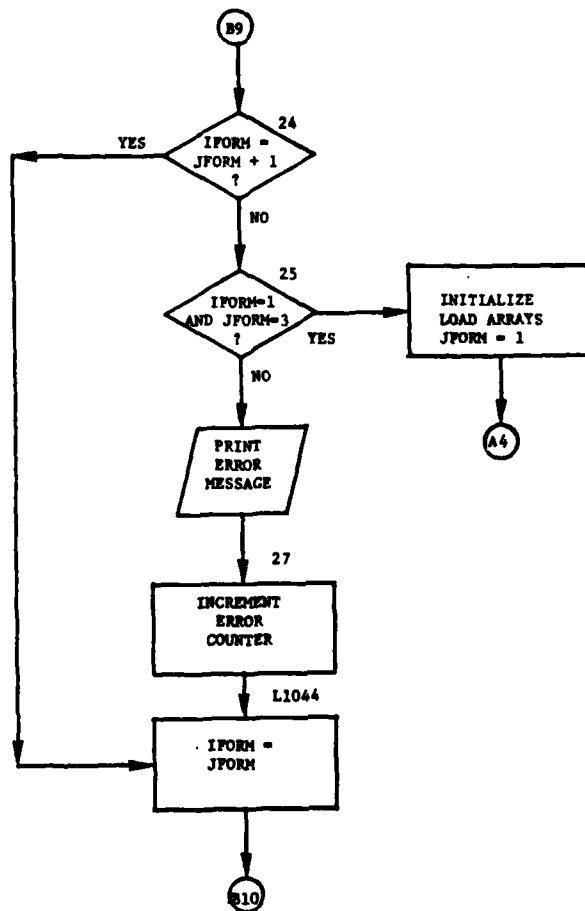


Figure II-10-B-2. Routine LOAD25 (Continued)

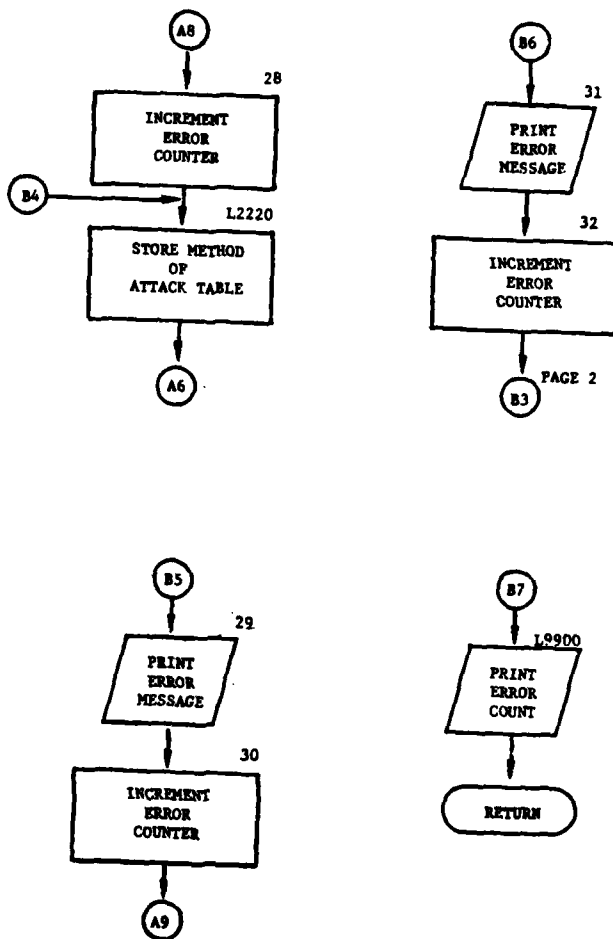


Figure II-10-B-2. Routine LOAD25 (Continued)

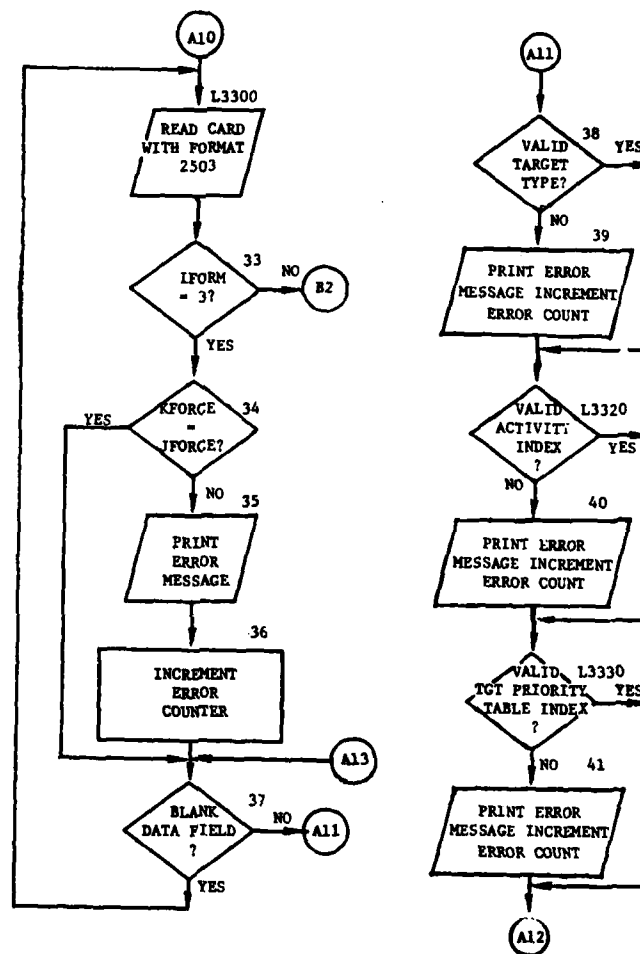


Figure II-10-B-2. Routine LOAD25 (Continued)

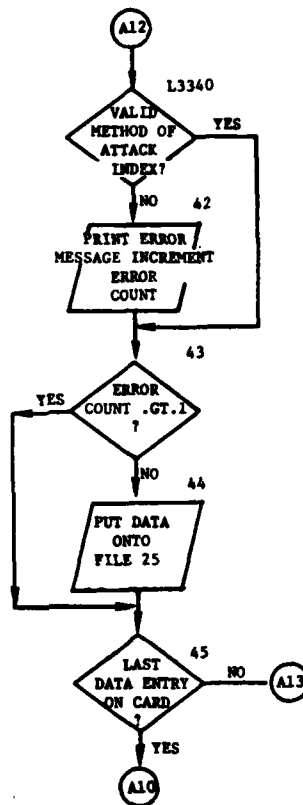


Figure II-10-B-2. Routine LOAD25 (Concluded)



(5) Blocks L300, 10, and L310. The second data card must have a valid force designator, R or B. If it does not, transfer control to block L210. The card contains the first set of target priority data.

(6) Blocks L1105, 11, and L1110. These blocks edit the card identification format, the force designator, and the target priority table index.

(7) Block L1120. The data on a single card, type 2501, contain two range cutoff values and three priority values (totaling five data entries) for each of four different target sizes. The data are stored in array IRNGP until all of the force's 2501 and 2502 card data have been processed. The data are then transferred to data file 25 using the indexing information contained on card format, identification 2503.

(8) Block L1100. The remainder of card format 2501 data for the force are read beginning with this block.

(9) Blocks L210, 12, 14, 15, 16, 17, 21, 22, 23, 26, 27, 28, 29, 30, 31, 32, L9900, 35, 36, 39, 40, 41, and 42. The logical process in each of these blocks prints an error message identifying the data error and increments an error counter index, NERR.

(10) Block L1033. Control transfers to the proper logic to process the card identification format. Card format identification groups 2501, 2502, and 2503 must be read in sequence for each respective force.

(11) Blocks 24 and 25. If the last card type read corresponds to the next sequence of the previously read card, control goes to block L1044. If the last card type read was a 2501 and the previous card was a 2503, control goes to block L1036; otherwise, control goes to block 26.

(12) Block L1036. Initialize load arrays and set JFORM equal to one. Control goes to block L300.

(13) Blocks L2200 through L2220. The method of attack table data for each weapon/munition combination listed are read into the MTHATK array for each method of attack table index. The data are stored in groups containing 10 data entries. Each card contains up to four groups.

(14) Blocks L3300 through 45. The logic represented in these blocks reads the data on the third format, 2503. Each set of data on this card format is required to associate a target combination with the correct target priority table and method of attack table. The data are transferred separately for each combination. If an error has been detected previously, none of the data are transferred to data file 25.

#### 4. ROUTINE OSORT:

a. Purpose. Routine OSORT is used to rank a list of N values in ascending order. The values are input in array LIST(N) in any order and are output in ranked order.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
LIST	Call	Array used to store values that are to be arranged.
N	Call	Total number of values in LIST array.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
LIST	Call	Array containing ranked packed priority values for each target combination.

d. Logical Flow (Figure II-10-B-3):

(1) Block 1. Set variable for the maximum number of passes to be made through array to be sorted.

(2) Block 2. Establish/increment outer do loop index. If all values have been sorted, return control to the calling routine.

(3) Blocks 3 and 4. Set inner loop scan limit and initialize MAX and INDEX.

(4) Block 5. Establish/increment inner do loop index. If inner do loop index is greater than the inner loop scan limit, control goes to block 6. Otherwise, control goes to block 8.

(5) Blocks 6 and 7. If the inner loop indexed value is less than MAX, control goes to block 5; otherwise, set MAX equal to that value and set INDEX equal to its index position.

(6) Block 8. If INDEX is equal to the inner loop scan limit, control goes to block 2.

(7) Block 9. Interchange the old and new maximum values. Control goes to block 2.

5. ROUTINE DUMP25

a. Purpose. Routine DUMP25 is called by routine TACLD to produce a printed output record of the data loaded onto data file 25 and a summarized listing of the TACFIRE Model's target priority tables for each force.

b. Input Variables: Input variables listed below are output variables in routine LOAD25.

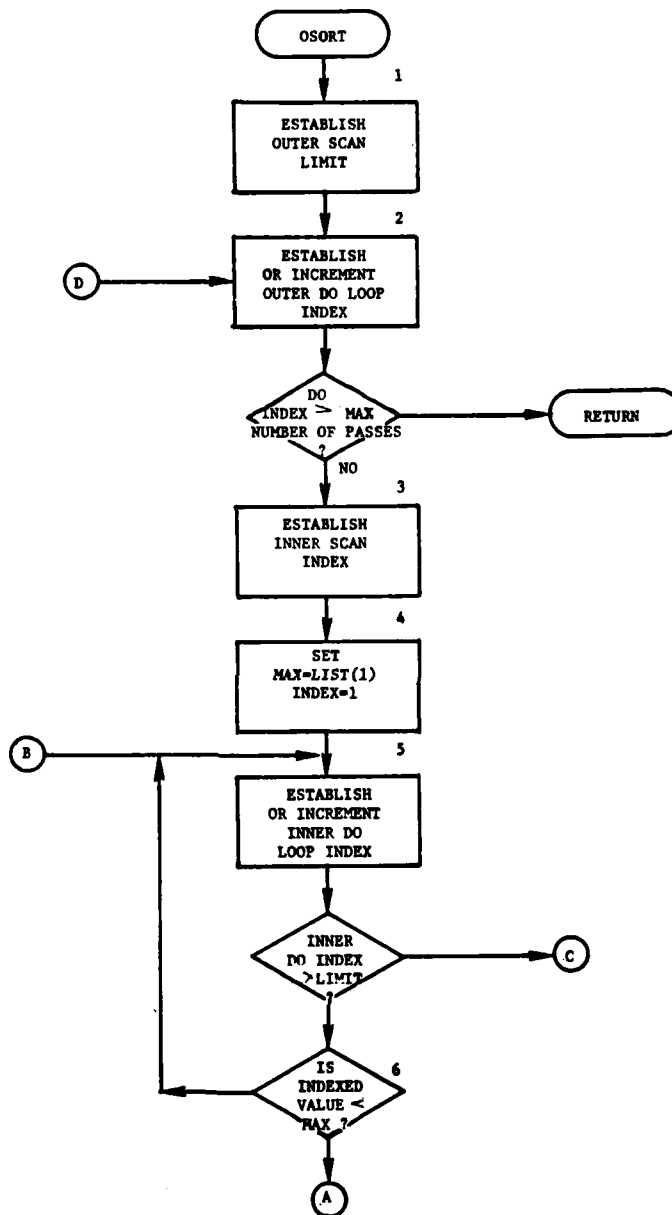


Figure II-10-B-3. Routine OSORT (Continued on Next Page)

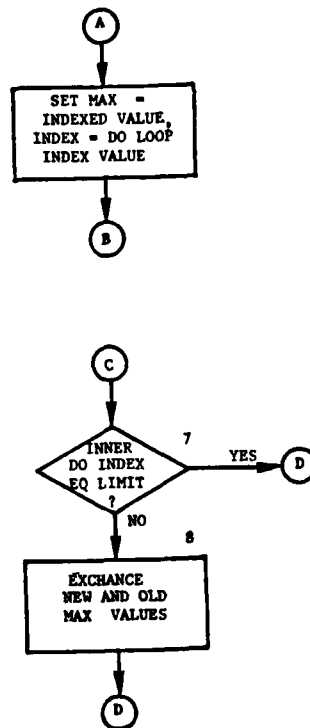


Figure II-10-B-3. Routine OSORT (Concluded)

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IRNGP(4,5,77)	DF25	Target priority tables.
MTHATK(20,5,77)	DF25	Method of attack tables.

c. Output Variables: The variables used as input are the output variables to the printer.

d. Logical Flow (Figure II-10-B-4):

- (1) Block 1. The labels used in each print header are initialized.
- (2) Block 2. Routine DUMP25 processes and lists output for each force. This block initializes or increments a loop to process each force.
- (3) Block L500. Routine GETWRD is called twice: once to retrieve the priority tables from data file 25 and once to retrieve the method of attack tables from data file 25.
- (4) Blocks 3 and 4. The do loop indexes to extract data from the target type and method of attack tables are initialized or incremented.
- (5) Blocks 5 and 6. The header and priority table are printed for this combination.
- (6) Block 7. A summarized listing of target combinations in order of priority is produced after the TACFIRE priority and method of attack tables have been listed for the force. To rank the data according to the priority value, the packed priority value consisting of the priority, type, activity, table index, and range indexes is computed. The packed priority value is in array IRANK. The original target priority scale will always be retained.
- (7) Block 8. A method of attack table is printed for the specific target type and target activity data being scanned.
- (8) Block L3000. If there are more activity types to be scanned, transfer control to block 3.
- (9) Block L4000. If there are more target types to be scanned, transfer control to block 4.
- (10) Block 9. Routine OSORT is called to rank the entries in array IRANK in ascending order.
- (11) Block 10. A summarized listing of target combinations, in order of decreasing priority, is printed as part of the TACLD output records.
- (12) Block L5000. If there is another force to be processed, transfer control to block 2; otherwise, return control to the calling routine.

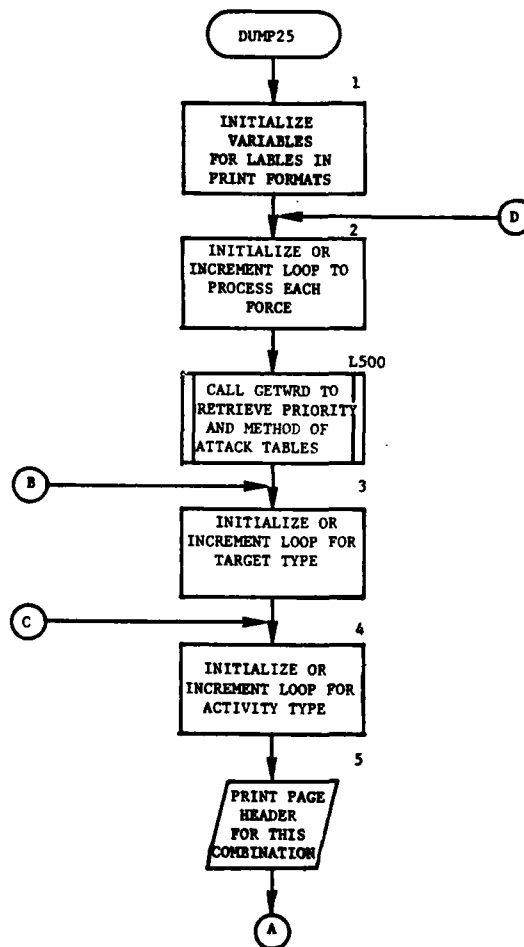


Figure II-10-B-4. Routine DUMP25 (Continued on Next Page)

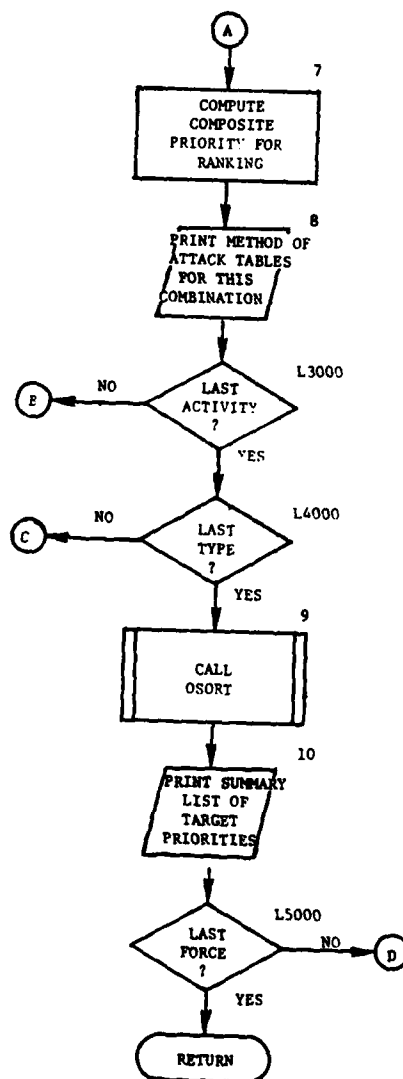


Figure II-10-B-4. Routine DUMP25 (Concluded)

NOTES

II-10-B-20



## APPENDIX C

### TACFIRE DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. Three types of printout formats are generated each time data are entered into the TACFIRE constant data input files. These printouts are listed in Figure II-10-C-1. These printouts were designed to assist those preparing constant data input to validate the data in the files and make necessary revisions as may be required.

Printout Title	Data Source
Data File 25 Data Card Image	Card data deck
Priority and Method of Attack	Data file 25
Summary Report of Target Priorities	Data file 25

Figure II-10-C-1. TACFIRE Constant Data Printouts

2. DATA FILE 25 DATA CARD IMAGE. The 80-column card image data is illustrated in Figure II-10-C-2. At the top of the figure at the far left is the number 1 followed by the letter B indicating that the card type is 1 and the force is Blue. At the far right is the card number 1, the sequence of that card within this subdeck; that is, it is the second card in the deck illustrated in the figure. The third column from the right has the card ID 2501, indicating that this is range priority type data. By referring to the discussion of card type 1 and ID 2501, the reader may determine the meaning of the other entries on this line. For example, the 011 at the left on the top line refers to the weapon munition index. For a complete description of the input card images, refer to Appendix A to Chapter 9 of the Period Processor, TACFIRE Model Input Requirements.

3. PRIORITY AND METHOD OF ATTACK TABLES. The second TACFIRE constant data input printout is labeled Priority and Method of Attack Tables. It is illustrated in Figure II-10-C-3. The headings on the various tables are similar to those for the card formats. At the top of the figure is shown the table of priority by target size and range to that target. These data are taken from card ID 2501. The lower half of the figure has the method of attack table and is a composite of information taken from cards 2502 and 2503. The order of choice of weapons with the weapon/munition index are arrayed against the number of volleys to be fired on each of the target sizes listed.

4. SUMMARY REPORT OF TARGET PRIORITIES. The last format of the printout displays the summary data on target priorities. It is illustrated in Figure II-10-C-4. The data headings in this table are those used in the card formats in describing the data entries. This table permits a review of the logic of priority and range choice.

CARD	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
CARD	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
CARD	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
CARD	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
CARD	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
CARD	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120
CARD	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140
CARD	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160
CARD	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180
CARD	181	182	183	184	185	186	187	188	189	190	191	192	193	194	195	196	197	198	199	200
CARD	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220
CARD	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240
CARD	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260
CARD	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279	280
CARD	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300
CARD	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320
CARD	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340
CARD	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360
CARD	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380
CARD	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400
CARD	401	402	403	404	405	406	407	408	409	410	411	412	413	414	415	416	417	418	419	420
CARD	421	422	423	424	425	426	427	428	429	430	431	432	433	434	435	436	437	438	439	440
CARD	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460
CARD	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480
CARD	481	482	483	484	485	486	487	488	489	490	491</									

Figure II-10-C-2. Data File 25 80-Column Card Image

[illegible]

BLUE PRIORITY AND METHOD OF ATTACK TABLES									
FOR RED		TARGETS OF TYPE		1 (INFANTRY)		IN ACTIVITY		1 (STAY )	
TARGET SIZE		PRIORITY (R(R1))		R1 (METERS)		PRIORITY BY TARGET SIZE AND RANGE TO TARGET		PRIORITY (R(R2))	

# SUMMARY REPORT OF TARGET PRIORITIES

## BLUE FIRING ON RED

PRIORITY	TARGET TYPE	TARGET ACTIVITY	TARGET SIZE	RANGE TO TARGET
1	ARTY G/H	FIRE	BN	10000 -- 30000
1	ARTY MSL	STAY	PLT	0 -- 5000
1	ARTY MSL	STAY	PLT	5000 -- 10000
1	ARTY MSL	STAY	PLT	10000 -- 999999
1	ARTY MSL	STAY	CO	0 -- 10000
1	ARTY MSL	STAY	CO	10000 -- 30000
1	ARTY MSL	STAY	CO	30000 -- 999999
1	ARTY MSL	STAY	BN	10000 -- 30000
1	ARTY MSL	STAY	BN	30000 -- 999999
1	ARTY MSL	FIPE	PLT	0 -- 5000
1	ARTY MSL	FIRE	PLT	5000 -- 10000
1	ARTY MSL	FIRE	PLT	10000 -- 999999
1	ARTY MSL	FIRE	CO	0 -- 10000
1	ARTY MSL	FIRE	CO	10000 -- 30000
1	ARTY MSL	FIRE	CO	30000 -- 999999
1	ARTY MSL	FIRE	BN	0 -- 10000
1	ARTY MSL	FIRE	BN	10000 -- 30000
1	ARTY MSL	FIRE	BN	30000 -- 999999
1	ADA GUNS	STAY	BN	0 -- 3000
1	ADA GUNS	STAY	BN	3000 -- 10000
1	ADA GUNS	MOVE	PLT	0 -- 3000
1	ADA GUNS	MOVE	PLT	3000 -- 5000
1	ADA GUNS	MOVE	CO	0 -- 3000
1	ADA GUNS	MOVE	CO	3000 -- 8000
1	ADA GUNS	FIRE	PLT	0 -- 3000
1	ADA GUNS	FIRE	PLT	3000 -- 5000
1	ADA GUNS	FIRE	CO	0 -- 3000
1	ADA GUNS	FIRE	CO	3000 -- 8000
1	ENGINEER	ENGINEER	CO	0 -- 3000
1	ENGINEER	ENGINEER	BN	0 -- 5000
2	INFANTRY	STAY	BN	0 -- 3000
2	INFANTRY	STAY	BN	3000 -- 10000
2	INFANTRY	MOVE	CO	0 -- 3000
2	INFANTRY	MOVE	CO	3000 -- 5000
2	INFANTRY	MOVE	BN	0 -- 3000
2	INFANTRY	MOVE	BN	3000 -- 10000
2	INFANTRY	ATTACK	PLT	0 -- 3000
2	INFANTRY	ATTACK	PLT	3000 -- 4000
2	INFANTRY	ATTACK	CO	5000 -- 999999
2	INFANTRY	ATTACK	BN	10000 -- 999999
2	INFANTRY	DEFEND	PLT	0 -- 3000
2	INFANTRY	DEFEND	PLT	3000 -- 4000
2	INFANTRY	DEFEND	BN	3000 -- 10000
2	INFANTRY	WITHDRAW	PLT	0 -- 3000
2	INFANTRY	WITHDRAW	BN	10000 -- 999999
2	ARMOR	MOVE	CO	0 -- 3000
2	ARMOR	MOVE	BN	0 -- 3000
2	ARMOR	MOVE	BOE	0 -- 3000

Figure II-10-C-4. Summary of Target Priorities

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR TACFIRE

(AVAILABLE UNDER SEPARATE COVER)

II-10-D-1

NOTES

II-10-D-2

## CHAPTER 11

### AIR GROUND ENGAGEMENT DATA INPUT PROGRAMS

1. INTRODUCTION. The Air Ground Engagement data input programs create and load the data files used to store the data required by the Air Ground Engagement Model. These data include tables of weapon descriptions and engagement results.

2. ROUTINES. The routine AIRLD comprises the Air Ground Engagement data input. It reads and edits the Air Ground Engagement data cards and loads the data on the appropriate file. After the load is complete, a formatted dump of each file is printed.

3. FILES. The following data files are used to store the Air Ground Engagement Model data:

- . Air Ground Engagement data - data file 26
- . Engagement results table - data file 27

NOTES

II-11-2



## APPENDIX A

### AIR GROUND ENGAGEMENT DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for air ground engagements are documented in Appendix A, Air Ground Engagement Model Input Requirements, to Chapter 10 of the Period Processor (Section IV).

NOTES

II-11-A-2

## APPENDIX B

### AIR GROUND ENGAGEMENT DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. This appendix provides the program description of routine AIRLD which loads the input data to the Air Ground Engagement Model. Routine SNATCH is a utility routine used to access the appropriate tables on data file 26. SNATCH is documented in Section VII.

#### 2. ROUTINE AIRLD:

a. Purpose. The AIRLD routine loads and dumps DIVWAG data files 26 and 27.

#### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ITASK	Card	Task to be performed (load or dump).
MT	Card	Type of mission that applies to data on the card.
IPRIOR	Card	Aircraft/munition mix.
ESTACT	Card	Estimated activity of target.
IAD	Card	Air defense code.
IFC	Card	Position in table where data will be put.
AMNT	Card	Number to be put in the table.
IPENN	Card	Penetration code.
IAC	Card	Aircraft item code.
IACMN	Card	Minimum number of aircraft required.
IACMX	Card	Maximum number of aircraft required.
IABRT	Card	Minimum number of aircraft required to perform the mission.
IFLMN	Card	Minimum amount of fuel required.
IFLMX	Card	Maximum amount of fuel required.
IGAGE	Card	Engagement range.
MIX	Card	Aircraft/munition mix.

<u>Name</u>	<u>Source</u>	<u>Contents</u>
MUN	Card	Munition item code.
ROUNDS	Card	Number of rounds being carried by the aircraft designated by mix.
IPERSN	Card	Number of personnel required.
ISLANT	Card	Average slant range.
IACC	Card	Accuracy of the weapon.
MIC	Card	Munition item code.
DAYATK	Card	Percent effectiveness of weapon in daytime when enemy attacks.
DAYDEF	Card	Percent effectiveness of weapon in daytime when enemy is defending.
NITATK	Card	Percent effectiveness of weapon at night when enemy attacks.
NITDEF	Card	Percent effectiveness of weapon at night when enemy is defending.
ISPEED	Card	Percent effectiveness of weapon at various aircraft speeds.
IALT	Card	Altitude code.
IRSPAR	Card	Percent effectiveness of weapon in rough terrain, sparse forest.
IRMED	Card	Percent effectiveness of weapon in rough terrain, medium forest.
IRDENG	Card	Percent effectiveness of weapon in rough terrain, dense forest, good intelligence.
IRDENP	Card	Percent effectiveness of weapon in rough terrain, dense forest, poor intelligence.
IGSPAR	Card	Percent effectiveness of weapon in good terrain, sparse forest.
IGMED	Card	Percent effectiveness of weapon in good terrain, medium forest.

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IGDENG	Card	Percent effectiveness of weapon in good terrain, dense forest, good intelligence.
IGDENP	Card	Percent effectiveness of weapon in poor terrain, dense forest, poor intelligence.
ITYPEA	Card	Average vulnerable area of aircraft to type A kill.
ITYPEB	Card	Average vulnerable area of aircraft to type B kill.
ITYPEC	Card	Average vulnerable area of aircraft to type C kill.
ITYPED	Card	Average vulnerable area of aircraft to type D kill.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
ACAVL	DF26	List of aircraft, munition, and weapon types.
TABLE	DF26	Data contained in Air Ground Engagement data file 26.
TABLE2	DF26	Data contained in Air Ground Engagement data file 26.
DATA	DF27	Engagement results table data for data file 27.

d. Logical Flow (Figure II-11-B-1):

- (1) Blocks 1 and 2. A call is made to routine GETFLE to retrieve the file name table and the first card is read.
- (2) Block 3. If the first card was a DUMP card control goes to block L479. If it was a LOAD card, processing continues.
- (3) Blocks 4 and 5. Data file 26 is created and initialized.
- (4) Block 6 and 7. Data file 27 is created and initialized.
- (5) Blocks 8, 9, and 10. Read the data cards and print card images and write card images onto a scratch file. When an end of data card is encountered rewind the scratch file.

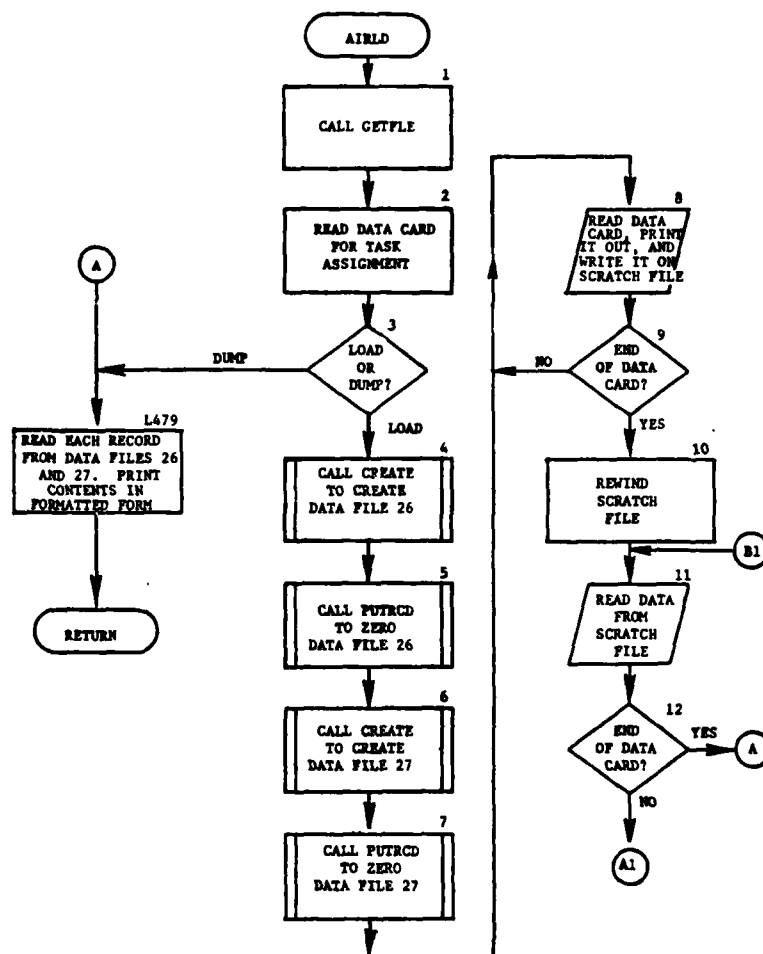


Figure II-11-B-1. Routine AIRLD (Continued on Next Page)

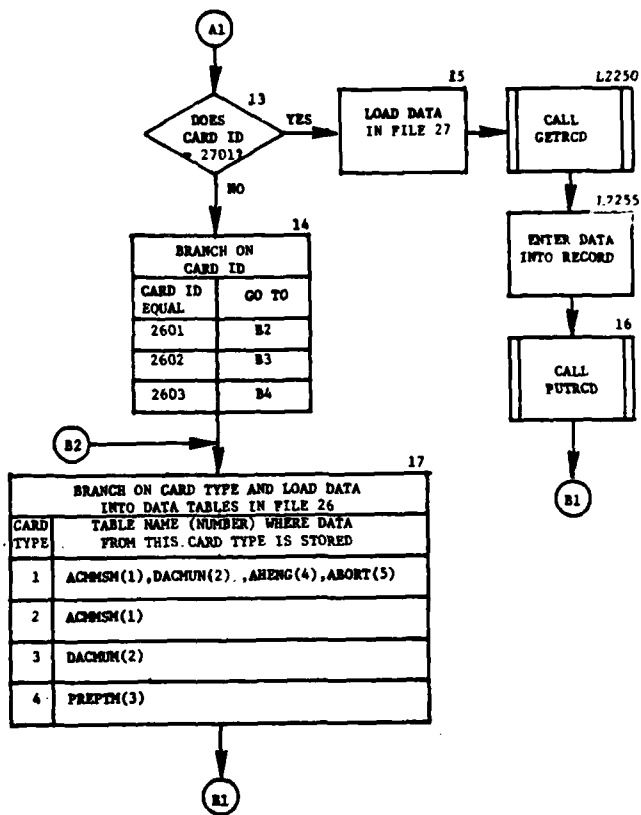


Figure II-11-B-1. Routine AIRLD (Continued)

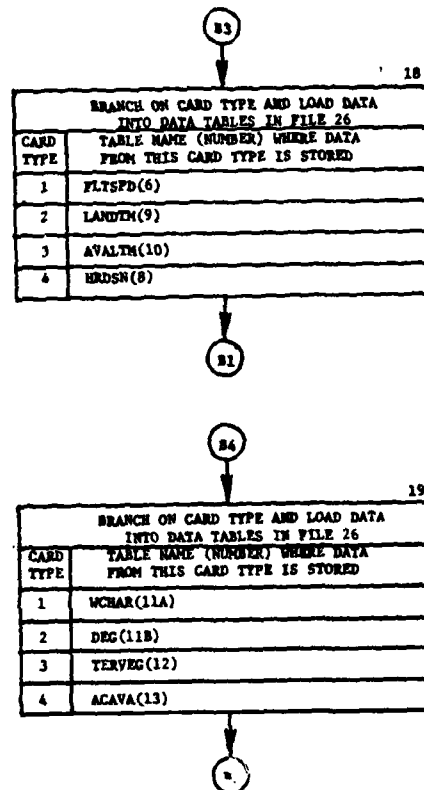


Figure II-11-B-1. Routine AIRLD (Concluded)



(6) Blocks 11 and 12. Read a card image from the scratch file. If an end of file was sensed control transfers to block L479.

(7) Block 13. If the data card ID is 2701 control goes to block 15; Otherwise, control goes to block 14.

(8) Block 14. Control transfers to block 17, 18, or 19 if the card ID is 2601, 2602, or 2603 respectively.

(9) Block 15. Determine the number of the data file 27 record to be loaded.

(10) Block L2250, L2255 and 16. Retrieve the record from data file 27, place the card data into the record, and replace the record onto data file 27. Control returns to block 11.

(11) Block 17. Place aircraft data in appropriate area of data file 26. Control transfers to block 11.

(12) Block 15. Place munition data in appropriate area of data file 26. Control returns to block 11.

(13) Block 19. Place air defense weapon data in appropriate area of data file 26. Control transfers to block 11.

(14) Block L479. Each record of data files 26 and 27 is retrieved and printed in formatted form. Control returns to the calling routine.

NOTES

II-11-B-8

## APPENDIX C

### AIR GROUND ENGAGEMENT DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. The 15 different printout formats listed in Figure II-11-C-1 are generated each time the constant data files of the Air Ground Engagement Model are recreated. These printouts are generated in the order in which they are listed. These printouts list all the data entered in the files and were designed for assistance to review and validate the data, and take remedial action if needed. In the figure are the titles of the printouts and the source from which these data were extracted. In most cases the data were taken from data file 26, but engagement results of air strikes were extracted from data file 27. The second column from the right lists the data card ID from which the information was initially read, and the last column on the right further identifies the individual card from which the basic data were extracted. Each of the printouts can thus be related to a specific file and, further, to the card that initially input the data.
2. DATA FILE 26 80-COLUMN CARD IMAGE. The first printout, Figure II-11-C-2, is a card image of the data that was read into data file 26. This printout consists of data that was punched into cards ID 2601, 2602, and 2603 in card types 1 through 4 in all three cases. Many of these card formats were produced in multiple copies to acquire and load all the pertinent data. The type of data entered into each of the card images that are pictured on these printouts may be interpreted from the columnar headings of the card format and by reference to Appendix A to Chapter 10, Air Ground Engagement Model Input Requirements, of the Period Processor.
3. AIRCRAFT AND MUNITIONS MINIMUM AND MAXIMUM REQUIREMENTS. The format of this printout is shown in Figure II-11-C-3. The aircraft and munition item codes and the minimum and maximum requirements of each are listed for the various mission types to be flown. There is also a listing for penetration missions for the Blue force and a similar set of listings for the Red force.
4. AIRCRAFT PREPARATION DELAY TIMES. The format of this printout is shown in Figure II-11-C-4. The time required to prepare an aircraft to fly a mission depends upon the number of aircraft flying the mission. The time is, therefore given in minutes for groups of five or less, for groups of five to ten, and for groups of more than 10 for all combinations of aircraft and mission types to be flown.
5. AIRCRAFT ENGAGEMENT PARAMETERS. The format of this printout is shown in Figure II-11-C-5. The aircraft engagement range is given in meters for each mission type that is to be flown by each of three classes of aircraft, two direct aerial fire support (DAFS) and one close air support (CAS).
6. AIRCRAFT MINIMUM ABORT DATA. The format of this printout is shown in Figure II-11-C-6. The number of aircraft required to successfully complete a mission is given for each class of aircraft that is to fly each mission. If fewer aircraft reach the target, the mission will be aborted.

Printout Titles	Data Source File	Card ID	Card Type
80-Column Card Image, Data File 26	Card data deck	All	All
Aircraft and Munitions Minimum and Maximum Requirements per Mission Type	Data file 26	2601	1
Aircraft Preparation Delay Times	Data file 26	2601	4
Aircraft Engagement Parameters	Data file 26	2601	1
Minimum Aircraft Abort Data	Data file 26	2601	1
Aircraft Landing Times	Data file 26	2602	2
Aircraft Flight Speeds	Data file 26	2602	1
Aircraft New Mission Availability Time	Data file 26	2602	3
Aircraft Degradation for Maintenance	Data file 26	2602	4
Aircraft Vulnerable Areas for Kills by Weapon Type	Data file 26	2603	4
Air Defense Weapon Characteristics	Data file 26	2603	1
Air Defense Weapons Effectiveness	Data file 26	2603	2
Air Defense Weapon Degradation from Terrain and Vegetation	Data file 26	2603	3
80-Column Card Images, Data File 27	Card data deck	All	All
Engagement Results of Air Strikes	Data file 27	2701	1

Figure II-11-C-1. Air Ground Engagement Constant Data Printouts

1A 13 88 2 4 1 162 2513000	98 4	1	1000	2601	1
1B 23 88 2 4 1 162 2513000	98 4	1	1000	2601	1
1C 53 88 2 4 1 162 2513000	98 4	1	1000	2601	1
1D 63 88 2 4 1 162 2513000	98 4	1	1000	2601	1
1E 93 88 2 4 1 162 2513000	98 4	1	1000	2601	1
1F 103 88 2 4 1 162 2513000	98 4	1	1000	2601	1
1G 133 88 2 4 1 162 2513000	98 4	1	1000	2601	1
1H 143 88 2 4 1 162 2513000	98 4	1	1000	2601	1
1I 173 88 2 4 1 162 2513000	98 4	1	1000	2601	1
1J 183 88 2 4 1 162 2513000	98 4	1	1000	2601	1
2B01TK CO DAY G0 VZ 1 90			2	2601	2A
2B02TK CO DAY PR VZ 1 90			2	2601	2B
2B05MEC IN DAY G0 VZ1 90			2	2601	2C
2B06MEC IN DAY PR VZ1 90			2	2601	2D
2B09ARTBTR DAY G0 VZ1 90			2	2601	2E
2B10ARTBTR DAY PR VZ1 90			2	2601	2F
2B13INF CO DAY G0 VZ1 7			2	2601	2G
2B14INF CO DAY G0 VZ1 7			2	2601	2H
2B17AFB DAY G0 VZ 1 90			2	2601	2I
2B18AFB DAY PR VZ 1 90			2	2601	2J
3B01TK CO DAY G0 VZ 1 90			2	2601	3A
3B02TK CO DAY PR VZ 1 90			2	2601	3B
3B05MEC IN DAY G0 VZ1 90			2	2601	3C
3B06MEC IN DAY PR VZ1 90			2	2601	3D
3B09ARTBTR DAY G0 VZ1 90			2	2601	3E
3B10ARTBTR DAY PR VZ1 90			2	2601	3F
3B13INF CO DAY G0 VZ1 7			2	2601	3G
3B14INF CO DAY PR VZ1 7			2	2601	3H
3B17AFB DAY G0 VZ 1 90			2	2601	3I
3B18AFB DAY PR VZ 1 90			2	2601	3J
4B01TK CO DAY G0 VZ 88			2601	4A	
4B02TK CO DAY PR VZ 88			2601	4B	
4B05MEC IN DAY G0 VZ 88			2601	4C	
4B06MEC IN DAY PR VZ 88			2601	4D	
4B09ARTBTR DAY G0 VZ 88			2601	4E	
4B10ARTBTR DAY PR VZ 88			2601	4F	

II-11-C-3

Figure II-11-C-2. Data file 26 80-Column Card Image

# BLUE AIRCRAFT AND MUNITIONS MIN/MAX REQUIREMENTS

## NON-PENETRATION MISSIONS

MIS. TYPE	A/C ITEM CODE	NO. A/C	MUNITIONS BY ITEM CODES (ROUNDS)										PERSONNEL		FUEL (GALS)	
			MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX
1	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
1	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
2	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
5	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
6	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	88 2	4	3	6	250	500	0	0	0	0	0	0	2	2	162	251
9	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	88 2	4	3	6	250	500	0	0	0	0	0	0	2	2	162	251
10	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
13	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
14	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
17	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
18	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

## PENETRATION MISSIONS

MIS. TYPE	A/C ITEM CODE	NO. A/C	MUNITIONS BY ITEM CODES (ROUNDS)										PERSONNEL		FUEL (GALS)	
			MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX
1	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
1	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
2	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
5	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
6	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	88 2	4	3	6	250	500	0	0	0	0	0	0	2	2	162	251
9	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	88 2	4	3	6	250	500	0	0	0	0	0	0	2	2	162	251
10	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251
13	98 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	88 2	4	4	6	250	500	0	0	0	0	0	0	2	2	162	251

Figure II-11-C-3. Aircraft and Munitions Minimum/Maximum Requirements Mission Type

# HLUE AIRCRAFT PREPARATION DELAY TIMES (MINUTES)

## NON-PENETRATION MISSIONS

MIS. TYPE	A/C TYPE	1 88	2 98	3 85	4 86	5 87	6 91	7 99	8 0	9 0
1	NO. A/C									
	0 - 5	5.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	5 - 10	5.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	NO. A/C									
	0 - 5	5.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	5 - 10	5.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	NO. A/C									
	0 - 5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	5 - 10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	NO. A/C									
	0 - 5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	5 - 10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	NO. A/C									
	0 - 5	5.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	5 - 10	5.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Figure II-11-C-4. Aircraft Preparation Delay Times

# BLUE AIRCRAFT ENGAGEMENT PARAMETERS

## NON-PENETRATION MISSIONS

	DAFS 1	DAFS 2	CAS
MISS. TYPE	RANGE (METERS)	RANGE (METERS)	RANGE (METERS)
1	3000.	0.	1000.
2	3000.	0.	1000.
3	0.	0.	0.
4	0.	0.	0.
5	3000.	0.	1000.
6	3000.	0.	1000.
7	0.	0.	0.
8	0.	0.	0.
9	3000.	0.	1000.
10	3000.	0.	1000.
11	0.	0.	0.
12	0.	0.	0.
13	3000.	0.	1000.
14	3000.	0.	1000.
15	0.	0.	0.
16	0.	0.	0.
17	3000.	0.	1000.
18	3000.	0.	1000.

Figure II-11-C-5. Aircraft Engagement Parameters



# BLUE AIRCRAFT MINIMUM ABORT DATA

## NON-PENETRATION MISSIONS

MIS. TYPE	DAFS 1	DAFS 2	CAS
1	1.	0.	1.
2	1.	0.	1.
3	0.	0.	0.
4	0.	0.	0.
5	1.	0.	1.
6	1.	0.	1.
7	0.	0.	0.
8	0.	0.	0.
9	1.	0.	1.
10	1.	0.	1.
11	0.	0.	0.
12	0.	0.	0.
13	1.	0.	1.
14	1.	0.	1.
15	0.	0.	0.
16	0.	0.	0.
17	1.	0.	1.
18	1.	0.	1.
19	0.	0.	0.
20	0.	0.	0.

## PENETRATION MISSIONS

MIS. TYPE	DAFS 1	DAFS 2	CAS
1	1.	0.	1.
2	1.	0.	1.
3	0.	0.	0.
4	0.	0.	0.
5	1.	0.	1.
6	1.	0.	1.
7	0.	0.	0.
8	0.	0.	0.
9	1.	0.	1.
10	1.	0.	1.
11	0.	0.	0.
12	0.	0.	0.
13	1.	0.	1.
14	1.	0.	1.
15	0.	0.	0.
16	0.	0.	0.

Figure II-11-C-6. Minimum Aircraft Abort Data

II-11-C-7

7. AIRCRAFT LANDING TIME. The format of this printout is shown in Figure II-11-C-7. The time required for an aircraft to land after it has entered the traffic pattern is given in minutes for each aircraft type according to the number of aircraft in the formation and the visibility conditions.
8. AIRCRAFT FLIGHT SPEEDS. The format of this printout is shown in Figure II-11-C-8. The average flight speed is given for each aircraft under four visibility conditions at cruise speed and under other specified conditions which that aircraft type would fly. The speeds are printed in knots.
9. AIRCRAFT NEW MISSION AVAILABILITY TIME. The format of this printout is shown in Figure II-11-C-9. The time required to make an aircraft available for a new mission depends upon the amount of damage incurred in the previous mission, or, if no damage, it depends upon the amount of time the aircraft was flown. Figure II-11-C-9 shows the time required for repairs after B-, C-, and D-kills, and the time to prepare an aircraft after a flight of less than 60 minutes and a flight of more than 60 minutes for each aircraft type to be flown. The times are in minutes.
10. AIRCRAFT DEGRADATION FOR MAINTENANCE. The format of this printout is shown in Figure II-11-C-10. The number of aircraft available for a mission is expressed as a decimal fraction of those that were available on the first day (D-Day). A number is shown for each aircraft type for a 14-day period, with the first day being D-Day.
11. AIRCRAFT VULNERABLE AREAS. The format of this printout is shown in Figure II-11-C-11. The numbers in the left-most column are the weapon types and the three two-digit numbers going across the page are the aircraft types. The four decimal numbers under the headings A, A+B, A+B+C, and A+B+C+D are the average vulnerable areas in square meters for each of the four kill categories when the specified aircraft type is fired on by the specified weapon type.
12. AIR DEFENSE WEAPON CHARACTERISTICS. The format of this printout is shown in Figure II-11-C-12. The average number of rounds fired per engagement, maximum effective range, average slant range and the average error are given for each weapon type under four different visibility conditions. The distance each weapon type is employed behind the FEBA is given in the last column. The weapon types are listed in the first column on the left.
13. AIR DEFENSE WEAPONS EFFECTIVENESS. The format of this printout is shown in Figure II-11-C-13. The column on the left is the weapon types and the next four columns indicate the percent of each of these weapons that would engage the target under each of the combinations of time of day and whether the enemy is attacking or defending. The remaining columns give the percent effectiveness of each weapon type when firing at targets moving at different speeds. The various speeds are given by the top number heading each column.
14. AIR DEFENSE WEAPON DEGRADATION FROM TERRAIN AND VEGETATION. The format of this printout is shown in Figure II-11-C-14. The column on the left is the weapon types. The percent of these weapons that are unaffected by the different conditions of terrain roughness, amount of vegetation, intelligence, and target altitude is given for each weapon type.

BLUE AIRCRAFT LANDING TIMES IN MINUTES FROM  
ENTRY INTO TRAFFIC PATTERN TO PARKED ON PAD

VISIBILITY	DAY - GOOD			DAY - POOR			NIGHT - GOOD			NIGHT - POOR		
	0-5	5-10	10 +	0-5	5-10	10 +	0-5	5-10	10 +	0-5	5-10	10 +
AIRCRAFT	0-5	5-10	10 +	0-5	5-10	10 +	0-5	5-10	10 +	0-5	5-10	10 +
A/C ITEM CODE												
88	10.	15.	20.	15.	20.	25.	15.	20.	25.	0.	0.	0.
98	5.	7.	10.	10.	20.	25.	10.	20.	25.	10.	20.	25.
85	10.	15.	20.	15.	20.	25.	15.	20.	25.	0.	0.	0.
86	10.	15.	20.	15.	20.	25.	15.	20.	25.	0.	0.	0.
87	10.	15.	20.	15.	20.	25.	15.	20.	25.	0.	0.	0.
91	10.	15.	20.	10.	15.	20.	15.	20.	25.	10.	20.	25.
99	5.	7.	10.	10.	20.	25.	10.	20.	25.	10.	20.	25.

Figure II-11-C-7. Aircraft Landing Times

BLUE AIRCRAFT FLIGHT SPEEDS (KNOTS)

A/C ITEM CODE	CRUISE SPEED		HELICOPTERS ONLY		RECON / SURV ONLY		CAS ONLY	
	DAY	NIGHT	ATTACKING TARGET	OVER TARGET	DAY	NIGHT	ATTACKING TARGET	NIGHT
VISX	GOOD	POOR	GOOD	POOR	GOOD	POOR	GOOD	POOR
88	130.	50.	50.	0.	110.	0.	0.	0.
98	440.	440.	440.	0.	0.	0.	1200.	1200.
85	119.	50.	50.	0.	110.	0.	0.	0.
86	92.	50.	50.	0.	110.	0.	0.	0.
87	155.	50.	50.	0.	110.	0.	0.	0.
91	225.	225.	225.	0.	0.	0.	0.	0.
99	440.	440.	440.	0.	0.	0.	1200.	1200.

Figure II-11-C-8. Aircraft Flight Speeds

TIME DELAY IN MINUTES BEFORE RETURNED BLUE  
AIRCRAFT ARE AVAILABLE FOR A NEW MISSION

A/C ITEM CODE	B - KILLS	C - KILLS	D - KILLS	UNDAMAGED A/C MISSION TIME (MIN) 0-60	60 +
88	555.	240.	120.	15.	15.
98	0.	0.	0.	0.	0.
85	555.	240.	120.	15.	15.
86	555.	240.	120.	20.	20.
87	555.	240.	120.	15.	15.
91	144.	720.	120.	20.	20.
99	0.	0.	0.	0.	0.

Figure II-11-C-9. Aircraft Availability Times for New Mission

II-11-C-11

DECIMAL FRACTION OF BLUE AIRCRAFT AVAILABLE  
AFTER SUBTRACTING THOSE DOWN FOR MAINTENANCE

A/C ITEM CODE	D-DAY + 1	D-DAY + 2	D-DAY + 3	D-DAY + 4	D-DAY + 5	D-DAY + 6	D-DAY + 7	D-DAY + 8	D-DAY + 9	D-DAY + 10	D-DAY + 11	D-DAY + 12	D-DAY + 13
88	1.000	.990	.950	.800	.800	.800	.800	.750	.750	.700	.600	.500	.400
98	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
85	1.000	.990	.950	.800	.800	.800	.800	.750	.750	.700	.600	.500	.400
86	1.000	.990	.950	.800	.800	.800	.800	.750	.750	.700	.600	.500	.400
87	1.000	.990	.950	.800	.800	.800	.800	.750	.750	.700	.600	.500	.400
91	1.000	.990	.950	.800	.800	.800	.800	.750	.750	.700	.600	.500	.400
99	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

Figure II-11-C-10. Aircraft Degradation for Maintenance

[illegible]

Figure II-11-C-11. Aircraft Vulnerable Areas for Kills by Weapon Type

# RED AIR DEFENSE WEAPON CHARACTERISTICS

WEPN ITEM CODE	ROUNDS FIRED PER ENGAGEMENT				MAXIMUM EFFECTIVE RANGE (METERS)				AVERAGE SLANT RANGE (METERS)				AVERAGE WEAPON ERROR (MILS)				DISTANCE EMPLOYED BEHIND FEB (METERS)	
	DAY		NIGHT		DAY		NIGHT		DAY		NIGHT		DAY		NIGHT		GOOD	POOR
	GOOD	POOR	GOOD	POOR	GOOD	POOR	GOOD	POOR	GOOD	POOR	GOOD	POOR	GOOD	POOR	GOOD	POOR		
22	49.	49.	16.	16.	1400.	800.	1400.	800.	370.	370.	370.	370.	40.	40.	40.	40.	500.	500.
70	273.	100.	73.	27.	2000.	800.	2000.	800.	1420.	450.	1420.	450.	10.	10.	10.	10.	1500.	1500.
71	784.	784.	784.	784.	3000.	3000.	3000.	3000.	1390.	1390.	1390.	1390.	10.	10.	10.	10.	3000.	3000.
72	26.	26.	26.	26.	800.	800.	3.	0.	1600.	800.	0.	0.	10.	10.	10.	10.	4000.	4000.
73	52.	52.	52.	52.	6000.	6000.	6000.	6000.	1990.	1990.	1990.	1990.	10.	10.	10.	10.	4000.	4000.
74	1.	1.	1.	1.	45000.	45000.	45000.	45000.	5000.	5000.	5000.	5000.	1.	1.	1.	1.	25000.	25000.
76	1.	1.	1.	1.	45000.	45000.	45000.	45000.	5000.	5000.	5000.	5000.	1.	1.	1.	1.	25000.	25000.
78	1.	1.	0.	0.	800.	800.	3.	0.	1060.	450.	8.	0.	1.	1.	0.	0.	600.	600.
79	3.	3.	3.	3.	35000.	35000.	35000.	35000.	5000.	5000.	5000.	5000.	1.	1.	1.	1.	17000.	17000.

Figure II-11-C-12. Air Defense Characteristics

II-11-C-14



# RED AIR DEFENSE WEAPONS EFFECTIVENESS DATA FACTORS

WEPN ITEM CODE	FRACTION DECIDING TO ENGAGE				FACTOR FOR MODIFYING ROUNDS FIRED PER ENGAGEMENT															
	DAY		NIGHT		AIRCRAFT SPEED (KNOTS)															
	ATK	DEF	ATK	DEF	40	60	80	100	120	140	160	180	200	220	240	260	280	300+		
22	50.0	50.0	3.0	3.0	99.9	99.9	95.0	93.0	90.0	80.0	60.0	50.0	40.0	35.0	30.0	25.0	25.0	25.0		
70	50.0	50.0	4.0	4.0	99.9	99.9	95.0	93.0	90.0	80.0	60.0	50.0	40.0	35.0	30.0	25.0	25.0	25.0		
71	50.0	50.0	50.0	50.0	99.9	99.9	95.0	93.0	90.0	80.0	60.0	50.0	40.0	35.0	30.0	25.0	25.0	25.0		
72	50.0	50.0	4.0	4.0	99.9	99.9	95.0	93.0	90.0	80.0	60.0	50.0	40.0	35.0	30.0	25.0	25.0	25.0		
73	50.0	50.0	50.0	50.0	99.9	99.9	95.0	93.0	90.0	80.0	60.0	50.0	40.0	35.0	30.0	25.0	25.0	25.0		
74	50.0	50.0	50.0	50.0	0.0	0.0	0.0	0.0	99.9	99.9	99.9	99.9	99.9	99.9	99.9	99.9	99.9	99.9		
76	50.0	50.0	50.0	50.0	0.0	0.0	0.0	0.0	99.9	99.9	99.9	99.9	99.9	99.9	99.9	99.9	99.9	99.9		
78	50.0	50.0	0.0	0.0	99.9	99.9	99.9	99.9	99.9	95.0	90.0	85.0	80.0	75.0	70.0	65.0	60.0	55.0		
79	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0		

Figure II-11-C-13. Air Defense Weapons Effectiveness

FACTORS FOR RED AIR DEFENSE WEAPON  
EFFECTIVENESS VS. TERRAIN AND VEGETATION

WEPN ITEM CODE	ALT (FEET)	GOOD TERRAIN				ROUGH TERRAIN			
		SPARSE	(FORESTATION)		SPARSE	MED	(FORESTATION)		DENSE
			GOOD INTEL	DENSE INTEL			GOOD INTEL	POOR INTEL	
22	NOF	.900	.800	.600	.950	.750	.500	.300	.300
22	1500	1.000	.950	.750	.950	.900	.750	.500	.500
22	6000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
70	NOF	.900	.800	.600	.950	.750	.500	.300	.300
70	1500	1.000	.950	.750	.950	.900	.750	.500	.500
70	6000	1.000	.950	.800	.950	.900	.750	.500	.500
71	NOF	.900	.800	.600	.950	.750	.500	.300	.300
71	1500	1.000	.950	.750	.950	.900	.750	.500	.500
71	6000	1.000	.950	.800	.950	.900	.750	.500	.500
72	NOF	.900	.800	.600	.950	.750	.500	.300	.300
72	1500	1.000	.950	.750	.950	.900	.750	.500	.500
72	6000	1.000	.950	.800	.950	.900	.750	.500	.500
73	NOF	.900	.800	.600	.950	.750	.500	.300	.300
73	1500	1.000	.950	.750	.950	.900	.750	.500	.500
73	6000	1.000	.950	.800	.950	.900	.750	.500	.500

II-11-C-16

Figure II-11-C-14. Air Defense Weapon Degradation from Terrain and Vegetation

15. ENGAGEMENT RESULTS. Two printouts deal with the results of air strikes, the 80-column card images of data file 27 and the engagement results of air strikes, which includes Blue aircraft striking Red targets and vice versa. The 80-column card image displays all data as originally entered in the card formats. For a description of the input cards, see Appendix A to Chapter 10, Air Ground Engagement Model Input Requirements, of the Period Processor. The engagement results of air strikes printout, Figure II-11-C-15, depicts the data entered into data file 27 and has been formatted with columnar headings and related information so that the data will be readily identifiable.

# ENGAGEMENT RESULTS FOR BLUE AIRCRAFT

ATTACKING RED TARGET

TIME = DAY  
 AIRCRAFT = 0AFK1  
 VISIBILITY = GOOD  
 ACTIVITY = ASSEMBLY AREA  
 TARGET = TANK COMPANY  
 AIR DEFENSE = NORMAL

TYPE OF KILL	WEAPON TYPE		WEAPON TYPE		WEAPON TYPE		WEAPON TYPE		WEAPON TYPE	
	SMALLER THAN 23MM	23MM	LARGER THAN 23MM	23MM	23MM	23MM	23MM	23MM	23MM	23MM
A-KILL	0.00000	.17847	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
P-KILL	0.00000	.01810	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
C-KILL	0.00000	.05590	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
D-KILL	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
TYPE 1 TYPE 2 TYPE 3 TYPE 4 TYPE 5 TYPE 6										
AERIAL MUNITION EXPENDED	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
AERIAL MUNITION LOST	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
TYPE 1 TYPE 2 TYPE 3 TYPE 4 TYPE 5 TYPE 6 TYPE 7 TYPE 8 TYPE 9										
AIR DEFENSE WEAPONS KILLED	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
AIR DEFENSE MUNITION EXPENDED	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

WEAPON	TANKS KILLED	APC KILLED	VEHICLES KILLED
GUIDED MISSILE	2.95000	.45000	0.00000
OTHER	0.00000	0.00000	0.00000

PERSONNEL KILLED = 14.10  
 NUMBER OF AIRCRAFT FLYING MISSION = 2.00  
 DURATION OF ENGAGEMENT IN MINUTES = 3.00

Figure II-11-C-15. Engagement Results of Air Strikes

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR AIR GROUND ENGAGEMENT

(AVAILABLE UNDER SEPARATE COVER)

II-11-D-1

NOTES

II-11-D-2

## CHAPTER 12

### SUPPRESSION DATA INPUT PROGRAMS

1. INTRODUCTION. The Suppression data input programs create and load the file containing the suppression group indexes and tables of suppression times used by the Suppression Model. These data are dependent upon data stored in data file 51 by the Organization and Equipment routines.

#### 2. ROUTINES:

a. General. The Suppression data input programs consist of the following routines.

b. Routine SPRSLD. This routine creates the Suppression data file, reads and edits the data cards, and stores the data in the file. It also associates the suppression group index input on the data cards with the UTDs using the table stored on data file 51.

c. Routine SUPDMP. This routine gives a formatted dump of the Suppression data loaded which includes the suppression groups and tables of suppression times associated with each group. It also generates an unformatted record dump of the file.

3. FILES. Data file 8 is loaded by the Suppression input program.

NOTES

II-12-2



## APPENDIX A

### SUPPRESSION DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for suppression are documented in Appendix A, Suppression Model Input Requirements, to Chapter 11 of the Period Processor (Section IV).

NOTES

II-12-A-2

## APPENDIX B

### SUPPRESSION DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. Suppression Model data are loaded by one routine, SPRSLD, which in turn calls routine SUPDMP to list the data. Since SPRSLD uses data file 51 as input, these data cannot be loaded until after an acceptable run of LDTOE.

#### 2. ROUTINE SPRSLD:

a. Purpose. This routine is used to load the constant data required by the Suppression Model. The data are read from card forms 0801 and 0802 and are stored on data file 8.

b. Input Variables. Data cards 0801 and 0802 and data file 51.

c. Output Variables. Tables to data file 8.

d. Logical Flow. (Figure II-12-B-1):

(1) Blocks 1, 2, and 3. The file name table is brought in from disk and data file 8 is released by a call to REMOVE and allocated by a call to CREATE with one record of 3200 words.

(2) Block 4. The list of unit type designators (UTD) required by this routine is retrieved from data file 51. These data must be loaded after the UTD list is finalized.

(3) Blocks L1000, 5, and L1125. The suppression groups (card types 0801) are read and stored in the array IFIL8, starting with word 1 if they are Blue force groups and with word 1001 if they are Red force groups. The 1000 locations reserved for each force correspond to the 1000 possible UTDs. The word location within the data file 8 record of the beginning of the suppression time table for the UTD associated group, is stored in the corresponding location in IFIL8.

(4) Blocks L2000, L2010, and L2040. The suppression time tables are read from card types 0802 and stored in IFIL8 starting in location 2001 if they are associated with the Blue force and in location 2601 if they are associated with the Red force. There are 12 words of data per table with a maximum of 50 tables. Each card contains an entire table. The suppression times are read in seconds and converted to and stored in centiminutes.

(5) Blocks L7000, 6, and 7. The suppression data are put into data file 8, and routine SUPDMP is called to give a word dump of data file 8. The file name table is printed.

#### 3. ROUTINE SUPDMP:

a. Purpose. This routine produces a dump of the contents of data file 8.

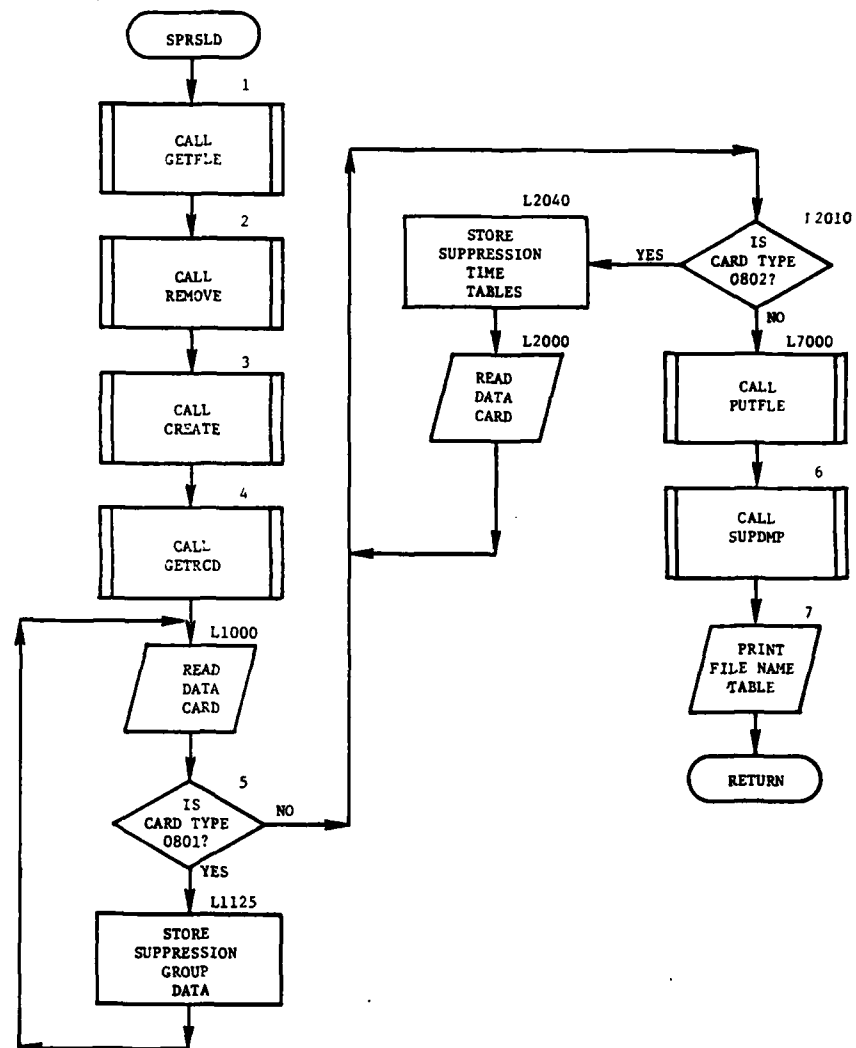


Figure II-12-B-1. Routine SPRSLD

- b. Input Variables. Tables on data file 8.
- c. Output Variables. Printed tables.
- d. Processing Description. The data loaded on data file 8 are read and dumped in file sequence.

NOTES

II-12-B-4

## APPENDIX C

### SUPPRESSION DATA LOAD OUTPUT DESCRIPTIONS

An 80-column image of each card is printed immediately after it is read. A sample of the listing produced in this manner is shown in Figure II-12-C-1. This data load program also provides a word dump of the file. This word dump may be used to verify that the data file was loaded correctly.





APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR SUPPRESSION

(AVAILABLE UNDER SEPARATE COVER)

II-12-D-1

NOTES

II-12-D-2

## CHAPTER 13

### NUCLEAR ASSESSMENT DATA INPUT PROGRAMS

1. INTRODUCTION. The Nuclear Assessment data input programs add the data required by the Nuclear Assessment Model to the data pertaining to the conventional Area Fire Model and already existing in data file 30. The controlling Nuclear load routine assures that the fire has been created. The data include tables describing the immediate time dependent effects of each nuclear detonation.

#### 2. ROUTINES:

a. General. The routines comprising the Nuclear Assessment data input programs are described in the following paragraphs.

b. Routine NUCLD. This routine controls the other routines for Nuclear data input. Its first task is to assure that the conventional Area Fire data are preserved. Once this is accomplished, routines FLD30 and DMP30 are called in that sequence.

c. Routine FLD30. This routine reads and edits the Nuclear Assessment data cards and stores the data on the file.

d. Routine CKARY. This is a utility routine which determines if a given integer is present in a given array.

e. Routine DMP30. This routine prints the nuclear data portion of the file in tabular form.

3. FILES. This program loads the part of data file 30 containing the nuclear data.

NOTES

II-13-2

## APPENDIX A

### NUCLEAR ASSESSMENT DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for nuclear assessment are documented in Appendix A, Nuclear Assessment Model Input Requirements, to Chapter 12 of the Period Processor (Section IV).

NOTES

II-13-A-2

AD-A112 650

ARMY COMBINED ARMS-COMBAT DEVELOPMENTS ACTIVITY FORT--ETC F/O 18/7  
DIWAS MODEL DOCUMENTATION. VOLUME II. PROGRAMMER/ANALYST MANUAL--ETC(U)  
JUL 76

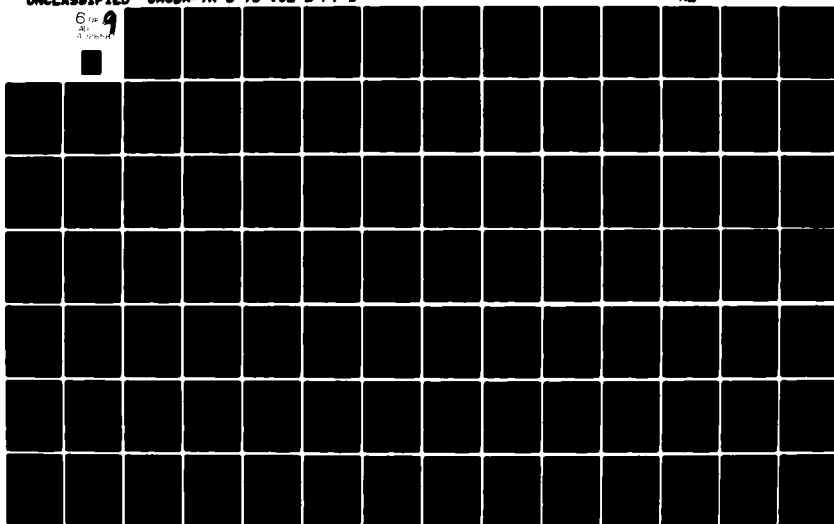
UNCLASSIFIED

CACDA-TR-8-76-VOL-2-PT-1

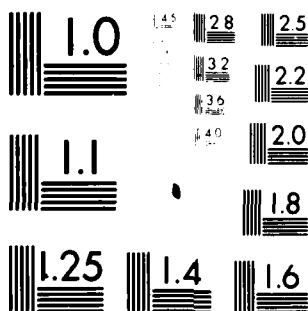
ML

6  
21  
21

9



11265



MICROCOPY RESOLUTION TEST CHART  
NBS 1963-A



## APPENDIX B

### NUCLEAR ASSESSMENT DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. Data for the Nuclear Assessment Model, when loaded, share data file 30 with data for the conventional Area Fire Model. The load programs are constructed in such a manner that the conventional Area Fire Model's data load routines must be executed prior to Nuclear Assessment Model data load routines. These routines include the controlling routine, NUCLD, that ensures proper allocation of data file 30 and maintenance of the conventional Area Fire Model data, FLD30, that reads the input data cards and stores data on data file 30, a utility routine, CKARY, and a dump routine, DMP30.

#### 2. ROUTINE NUCLD:

a. Purpose. NUCLD is the driving routine for FLD30. It copies the two records on data file 30, recreates data file 30 with 21 records, and puts the first two records back on the new data file 30.

b. Input Variables. None.

c. Output Variables:

(1) Data file 30.

(2) Other variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
LUCR	ONE	Logical unit number of card reader.
LUPR	ONE	Logical unit number of printer.
LUIN	ONE	File number of intermediate file.

#### d. Logical Flow (Figure II-13-B-1):

(1) Block 1. Initialize the numbers of the devices to be used by FLD30 for input and output operations.

(2) Blocks 2 and 3. Bring in the file name table and print it.

(3) Blocks 4 and 5. Copy the first two records of data file 30 and call REMOVE to remove it.

(4) Blocks 6 and 7. Call CREATE to recreate data file 30 and copy the two records from the original data file 30 onto it.

(5) Block 8. Initialize the added records on data file 30 as either blank or zero.

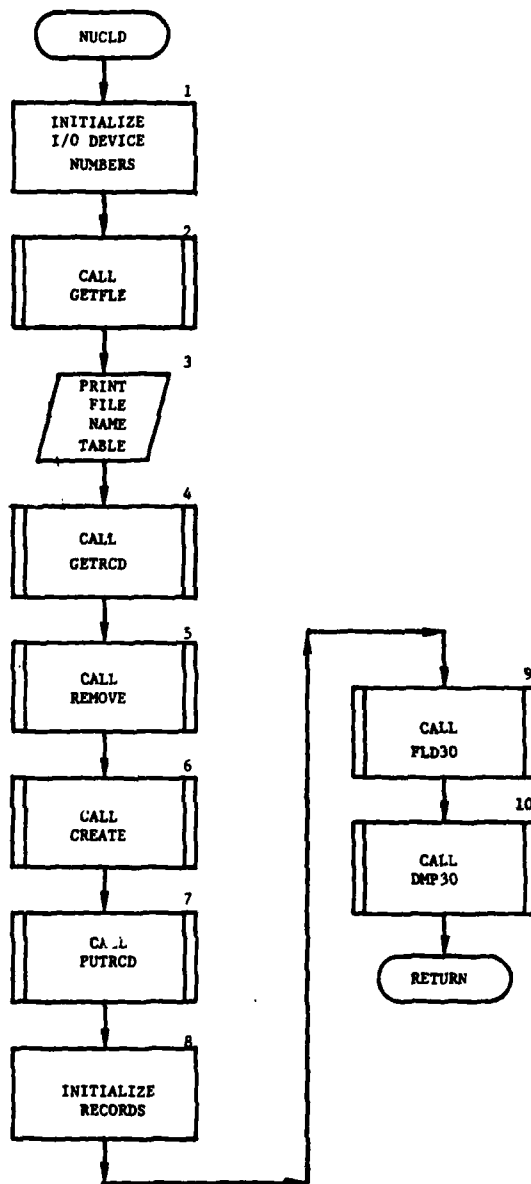


Figure II-13-B-1. Routine NUCLD

(6) Blocks 9 and 10. Call FLD30 to read and edit the input data and store it on data file 30. Call DMP30 to print the contents of data file 30.

3. ROUTINE FLD30:

a. Purpose. FLD30 reads and edits data for the Nuclear Assessment Model and stores it on data file 30.

b. Input Variables. Input variables are the output variables of NUCLD (paragraph 2c).

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
FAYT(7,1200)	DF30	Fuze data for nuclear weapons.  FAYT(1,I). Code number for the weapon-fuze combination.  FAYT(2,I). Impact detonation flag.  FAYT(3,I). Airburst option flag.  FAYT(4,I). Desired height of burst if the airburst flag is on.  FAYT(5,I). Probable error in height of burst.  FAYT(6,I). Minimum range for this weapon-fuze combination.  FAYT(7,I). Maximum range for this weapon-fuze combination.
NOBFAC(30)	DF30	Character codes to link barriers to the array of pointers to damage radii.
NOBFPT(30)	DF30	Array of pointers linking barriers to damage radii.
RT(9)	DF30	Radiation modifiers and soil type indicator.
WWNAME(1200)	DF30	Array name assigned to nuclear weapon-warhead-fuze combinations.  WWSTAT(9,100). Array of weapon-warhead statistics.  WWSTAT(1,I). Code number for the weapon-warhead combination.

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
		WWSTAT(2,I). Equipment item code for the weapon.
		WWSTAT(3,I). Equipment item code for the warhead.
		WWSTAT(4,I). Standard deviation in range for the weapon-warhead combination.
		WWSTAT(5,I). Standard deviation in deflection for the weapon-warhead combination.
		WWSTAT(6,I). Index number of the FAYT array of the first entry for this weapon-warhead combination.
		WWSTAT(7,I). Incoming angle of the round.
		WWSTAT(8,I). Velocity of the round (meters per minute).
		WWSTAT(9,I). Time required to deliver from state of readiness one.
YDRHOB(241,30)	DF30	Table associating, with each yield, four heights of burst and radii choices for each of the 30 possible damage radii.
EOHLEA(200)	DF30	Array of links from equipment item codes of items in the air to damage radii of enemy weapons.
EOHLFA(200)	DF30	Array of links from equipment item codes of items in the air to damage radii of friendly weapons.
PEPFLG(200)	DF30	Array of flags to show troop carrying items.
CCYD(30)	DF30	Codes of the yields in the YDRHOB table.
UNPRO(70)	DF30	Array of distributions of unprotected personnel.
		UNPRO(1,I). Personnel exposed (unwarned).
		UNPRO(2,I). Personnel in open foxholes (unwarned).
		UNPRO(3,I). Personnel in earth shelters (unwarned).

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
		UNPRO(4,I). Personnel exposed (warned).
		UNPRO(5,I). Personnel in open foxholes (warned).
		UNPRO(6,I). Personnel in earth shelters (warned).
		UNPRO(7,I). Time to revert to unwarned status.
EOHLEG(200)	ONE	Array of links from equipment item codes of items on the ground to damage radii of enemy weapons.
EOHLFG(200)	ONE	Array of links from equipment item codes of items on the ground to damage radii of friendly weapons.
RSTRAN(200)	ONE	Array of residual transmission factors for each equipment item.

d. Logical Flow (Figure II-13-B-2):

- (1) Blocks L700 and 1. Initialize the arrays to be used for storage of data before it is transferred to data file 30. If the input is for the Red force, transfer control to block 8.
- (2) Block 2. Read the data cards and print each one on the printer.
- (3) Blocks 3 and 4. As the cards are read, store card image on an intermediate file from which they will again be read for editing and storing into arrays.
- (4) Blocks 5, 6, 7, and 8. Read the first data card image and store the radiation modifiers and soil type indicator in the output array. Read the next card image. Control branches on the appropriate card type.
- (5) Blocks L10, 10, 11, 12, 13, and 14. Process the type one card images by reading them from the intermediate file and checking for redundant cards or values not within range. Print error messages.
- (6) Block 15. Put the edited data on data file 30.
- (7) Blocks L38, 16, 17, L30, 18, 19, L40, 20, 21, L50, 22, 23, L60, 24, and 25. Process other card types according to their associated formats, checking for redundant cards or values not within range.
- (8) Block 26. Store the data from card types two through six on data file 30.

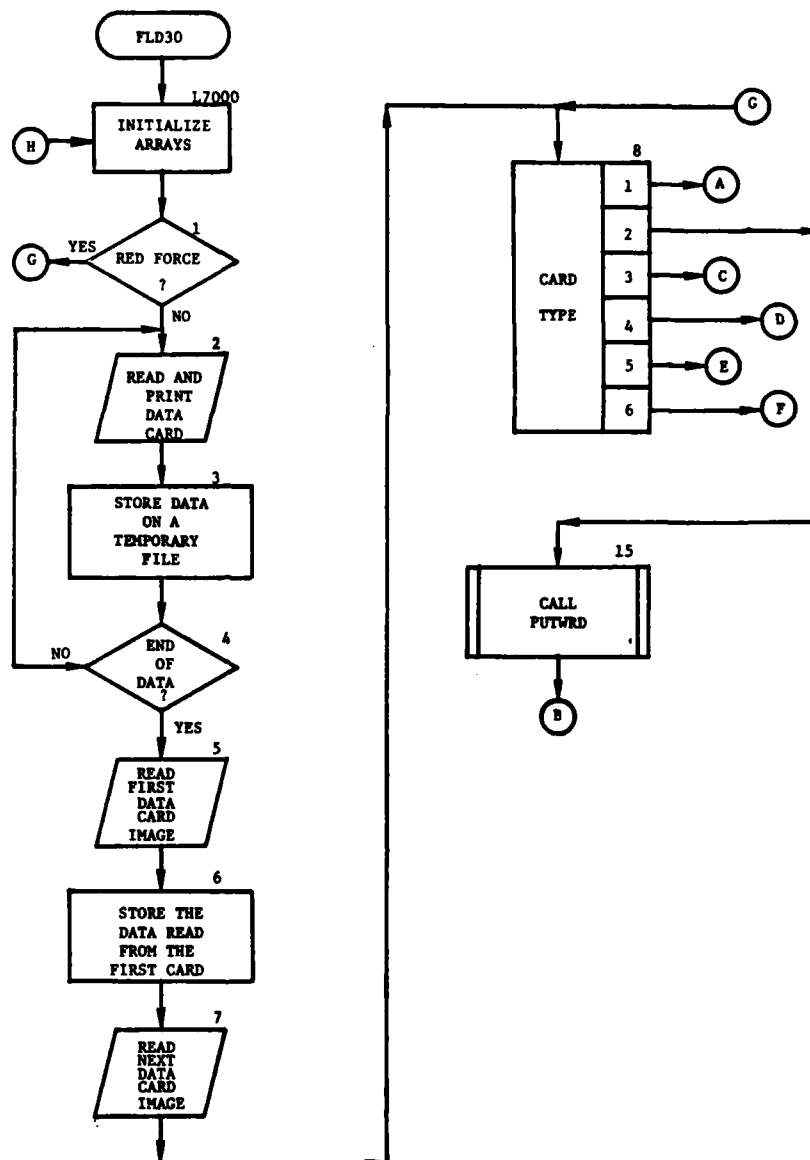


Figure II-13-B-2. Routine FLD30 (Continued on Next Page)

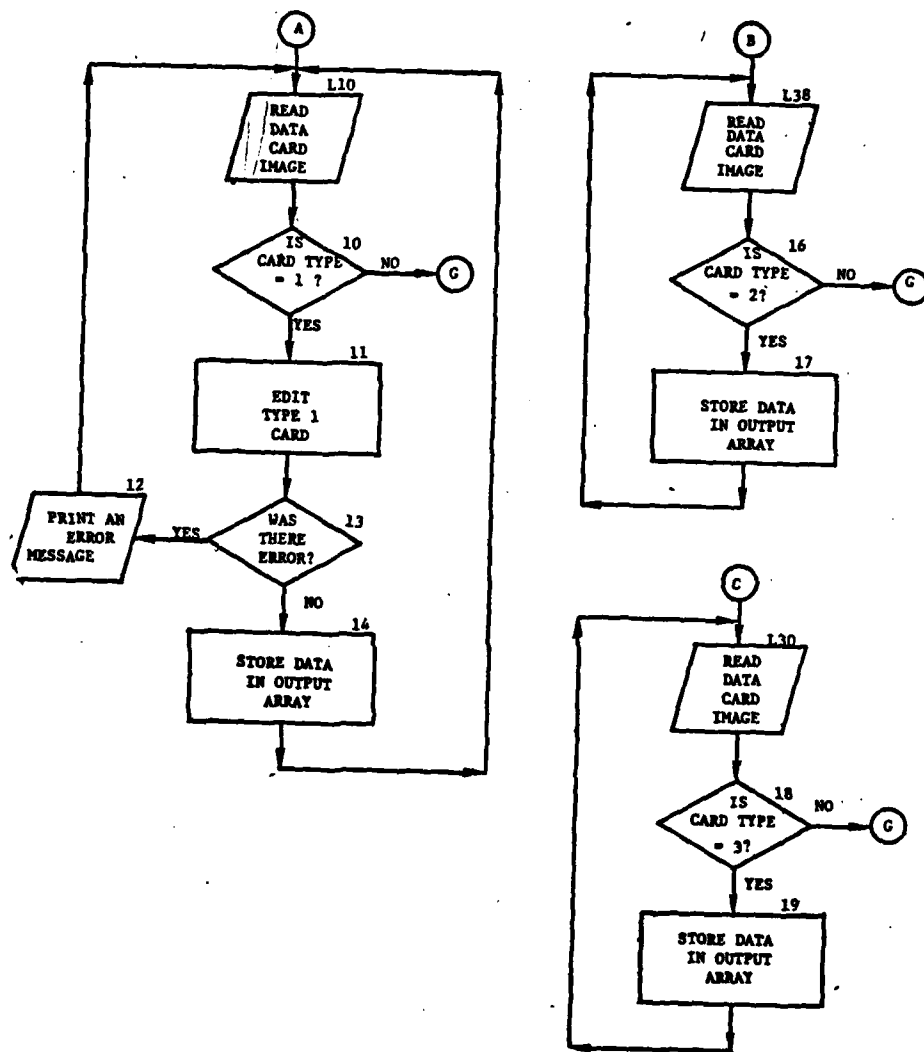


Figure II-13-B-2. Routine FLD30 (Continued)

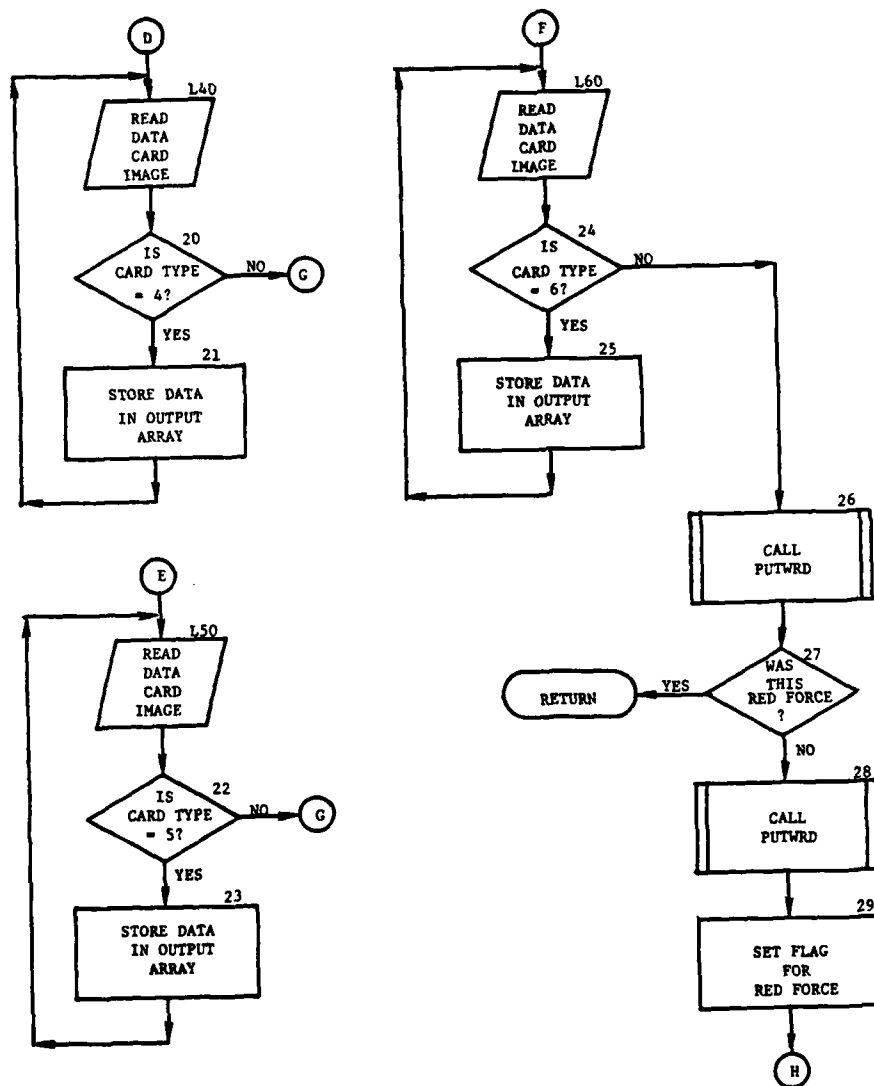


Figure II-13-B-2. Routine FLD30 (Concluded)



(9) Block 27. If data for both forces have been processed, return control to the calling routine.

(10) Blocks 28 and 29. Put data on data file 30 and set a flag to indicate that the Red force data is being entered. Control goes to block L700 to process the Red force data.

#### 4. ROUTINE CKARY:

a. Purpose. CKARY searches an integer array for a special number. If the number is not found, a zero is returned; if the number is found, a one is returned.

##### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IARRY	Call	Array to be searched.
LOOK	Call	Number to find in IARRAY.
MAX	Call	Maximum number of elements to search in the array IARRY.

##### c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
IANS	Call	Flag indicating whether the number was in the array.

##### d. Logical Flow (Figure II-13-B-3):

(1) Block 1. Initialize the flag to zero.

(2) Block 2. Loop through IARRY from one to MAX, searching for the number specified. If the number is not found, return control to the calling routine.

(3) Block L2. If the specified number was found, set the flag to one and return control to the calling routine.

#### 5. ROUTINE DMP30.

a. Purpose. DMP30 prints the contents of data file 30 in formatted tabular form.

b. Input Variables. Contents of data file 30.

c. Output Variables. Formatted tables printed with contents of data file 30.

d. Processing Description. Each type of data is read from data file 30, reformatted into easily readable tables, and printed on the line printer.

II-13-B-10

NOTES

II-13-B-12

## APPENDIX C

### NUCLEAR ASSESSMENT DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. Ten different printouts are generated each time data are entered in the Nuclear Assessment constant data file. These printouts are listed in Figure II-13-C-1.

Printout Title	Date Source	Card ID
80-Column Card Image	Card Data Deck	All
Weapon Combination Names	Data file 30	3011
Weapon Statistics	Data file 30	3011
Fuze Table Entries	Data file 30	3011
Yield Table	Data file 30	3011
EOH Link to Damage Radii	Data file 30	3011
EOH Data	Data file 30	3011
Yield Codes and Barrier Data	Data file 30	3011
Unprotected Personnel Distribution	Data file 30	3011
Special Factors for Both Teams	Data file 30	3011

Figure II-13-C-1. Nuclear Assessment Model Constant Data Input Printouts

2. NUCLEAR ASSESSMENT 80-COLUMN CARD IMAGE. The 80-column card image is printed so that the data entries are in the same format as entered in the card type. The format for the 80-column card image is illustrated in Figure II-13-C-2. On the top line at the far left is the number 7, indicating card type 7 for both forces. In the third column from the right is the ID number 3011.

[illegible]

**Figure II-13-C-2. Nuclear Assessment 80-Column Card Image**

3. WEAPON COMBINATION NAMES. The format for this report, described by its title, is illustrated in Figure II-13-C-3 for the Blue team. A similar report is produced for the Red team.
4. WEAPON STATISTICS. This report, Figure II-13-C-4, identifies the weapon statistics for each combination of weapon delivery system and weapon item code. Column one identifies each such combination. The next eight columns identify respectfully the weapon item code, warhead item code, the range dispersion, deflection dispersion, weapon system index, delivery angle, projectile velocity, and delay time associated with each weapon/delivery combination.
5. FUZE TABLE ENTRIES. This report, Figure II-13-C-5, identifies fuze characteristics for each type of fuze in column one. The next six columns identify respectfully the impact option indicator, height of burst option, preset height of burst, probable error in height of burst, and the minimum and maximum range of dispersion associated with each fuze type.
6. YIELD TABLE. This report, Figure II-13-C-6, reflects the height of burst and damage radius for each damage index of each weapon system.
7. EOH LINKS. This report, Figure II-13-C-7, associates equipment with the appropriate assessment radii of effect. This assessment is given for friendly and enemy munitions both airborne and on the ground.
8. EOH DATA. This report, Figure II-13-C-8, reflects the radiation transmission factor and the personnel protection flag for each equipment type.
9. YIELD CODES AND BARRIER DATA. This report, illustrated in Figure II-13-C-9, reflects barrier damage by yield code and damage radius index.
10. UNPROTECTED PERSONNEL DISTRIBUTION. This report, Figure II-13-C-10, reflects the percentage of personnel killed in three different postures, engaged in seven different activities, for both warned and unwarned. It also gives the time for personnel engaged in the seven activities to revert back to an unwarned condition.
11. SPECIAL FACTORS FOR BOTH TEAMS. The first two entries in this report, Figure II-13-C-11, give the slant range in yards corresponding to a radiation rate of 0.1 rads/hour/ton, the slant range in yards corresponding to a radiation rate of 0.005 rads/hour/ton. The next four entries give the same data for kiloton and megaton yields. The next entry is the soil type multiplier. The next entry is zero--and meaningless--and the last entry is minutes to resume unwarned posture for stay activity.

NSAB	NCAC	NCAD	NRAE	NRAF	NRAH	NRIF	NRIH	NLAG	NLBG	NLIG	NMAI	NMBI	NMII
NMAJ	NMBJ	NMIJ	NHAK	NHBK	NHIK	NABG	NABL	NAIL	NABN	NAIN	DDSA	DDSF	

**Figure II-13-C-3. Weapon Combination Names**

[illegible]

**Figure II-13-C-4. Weapon Statistics**

THE FUZE TABLE ENTRIES FOR BLUE TEAM.							
CODE	IMPACT	HQX FLAG	HQX	PEH	MIN RANGE	MAX RANGE	
A	0	0	-0	15	2000	10000	
A	0	0	-0	28	2000	20000	
A	0	0	-0	28	2000	20000	
A	0	0	-0	140	5000	40000	
A	0	0	-0	140	5000	40000	
A	0	0	-0	140	5000	40000	
I	1	0	-0	-0	5000	40000	
I	1	0	-0	-0	5000	40000	
I	1	0	-0	-0	5000	40000	
A	0	1	19	30	10000	50000	
A	0	2	39	30	10000	50000	
B	1	1	19	30	10000	50000	
B	1	2	39	30	10000	50000	
I	1	0	-0	-0	10000	50000	
A	0	1	32	50	50000	150000	
A	0	2	66	50	50000	150000	
B	1	1	32	50	50000	150000	
B	1	2	66	50	50000	150000	
I	1	0	-0	-0	50000	150000	
A	0	1	50	50	50000	150000	
A	0	2	79	50	50000	150000	
B	1	1	50	50	50000	150000	
B	1	2	79	50	50000	150000	
I	1	0	-0	-0	50000	150000	
A	0	1	80	100	50000	300000	
A	0	2	118	100	50000	300000	
B	1	1	80	100	50000	300000	
B	1	2	118	100	50000	300000	
I	1	0	-0	-0	50000	300000	
B	1	1	26	50	0	999999	
B	1	2	46	50	0	999999	
I	1	0	-0	-0	0	999999	
B	1	1	74	50	0	999999	
B	1	2	123	50	0	999999	
I	1	0	-0	-0	-0	-0	
B	1	1	131	50	-0	-0	
B	1	2	215	50	-0	-0	
I	1	0	-0	-0	-0	-0	
S	1	0	-0	-0	-0	-0	
S	1	0	-0	-0	-0	-0	
	0	0	0	0	0	0	
	0	0	0	0	0	0	
	0	0	0	0	0	0	
	0	0	0	0	0	0	
	0	0	0	0	0	0	
	0	0	0	0	0	0	
	0	0	0	0	0	0	
	0	0	0	0	0	0	
	0	0	0	0	0	0	
	0	0	0	0	0	0	

Figure II-13-C-5. Fuze Table Entries



[illegible][illegible]

FEED=		1.00													
50	500	573	333	530	453	400	50	410	200	790	300	760	450	680	
50	450	413	333	340	450	50	50	643	230	610	300	560	450	450	
50	270	200	193	300	0	450	0	420	230	370	300	290	450	0	
50	560	500	570	300	470	450	330	50	760	200	740	300	450	623	
50	440	203	403	330	370	450	0	50	630	200	600	300	550	440	
50	1000	200	1000	350	1300	450	1000	50	600	200	600	300	400	600	
50	500	500	503	300	500	450	500	50	250	200	250	300	250	250	
50	130	260	300	300	0	450	0	50	1020	200	1250	300	1350	1160	
50	230	203	133	300	0	450	0	50	520	200	573	300	710	450	0
50	290	300	200	300	0	450	0	50	160	200	0	300	0	450	0
50	330	200	240	310	0	450	0	50	700	200	660	300	600	450	450

II-13-C-6



EOH DATA FOR BLUE TEAM.		
RESIDUAL TRANS. FACTOR		PERSONNEL PROTECTION FLAG
10.0000		0
10.0000		0
10.0000		0
10.0000		0
10.0000		0
10.0000		0
10.0000		0
10.0000		0
10.0000		0
10.0000		0
0.0000		0
0.0000		0
0.0000		0
0.0000		0
0.0000		0
0.0000		0
0.0000		0
0.0000		0
0.0000		0
10.0000		0
10.0000		0
10.0000		0
0.0000		0
10.0000		0
7.0000		2
10.0000		0
10.0000		0
10.0000		0
7.0000		2
10.0000		0
10.0000		0
10.0000		0
10.0000		0
7.0000		2
7.0000		2
10.0000		0
3.0000		1
10.0000		0
10.0000		0
10.0000		0
3.0000		1
10.0000		0
10.0000		0
10.0000		0
0.0000		0
0.0000		0
0.0000		0
0.0000		0

Figure II-13-C-8. EOH Data



UNPROTECTED PERSONNEL DISTRIBUTION FOR BLUE TEAM.						
STANDING	UNARMED FOXHOLES	EARTH SHELTERS	STARTING	WARNED FOXHOLES	EARTH SHELTERS	TIME TO REVEAL TO UNARMED
30	5	5	5	85	15	60
70	15	15	5	50	45	350
50	20	20	0	40	60	723
0	0	0	0	0	0	0
100	0	0	20	75	5	5
70	15	15	5	50	45	10
100	0	0	60	40	50	2
30	40	30	10	40	10	5
90	5	5	5	85	10	5
100	0	0	50	40	10	4

Figure II-13-C-10. Unprotected Personnel Distribution

SPECIAL FACTORS FOR BLUE TEAM

.57287  
.01352  
2264.62500  
29807.53444  
.00586  
12.00000  
0.00000  
5.00000

Figure II-13-C-11. Special Factors

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR NUCLEAR ASSESSMENT

(AVAILABLE UNDER SEPARATE COVER)

II-13-D-1

NOTES

II-13-D-2

## CHAPTER 14

### MOVEMENT DATA INPUT PROGRAMS

1. INTRODUCTION. The Movement data input routines create and load the data files required to contain the data associated with the Movement Model. They require, as input, data file 51 created by the Organization and Equipment data input programs. The data stored include tables of mobility categories, mobility classes, movement rates, and fuel consumption rates.

#### 2. ROUTINES:

a. General. The Movement input programs consist of the following routines.

b. Routine MOVELD. This routine creates the files required, reads and edits the data cards, and loads the data in the files. It also calls routine DPMOVE. It uses the unit type designator (UTD) directory table from data file 51 to associate UTDs with mobility categories.

c. Routine DPMOVE. This routine provides a formatted dump of the files created in tabular form. An unformatted record dump of each file is also given.

#### 3. FILES. The files created by this program are:

. Mobility category tables	- data file 9
. Exclusion and miscellaneous tables	- data file 14
. Fuel consumption rate tables	- data file 15
. Movement rate tables	- data file 19



NOTES

II-14-2

## APPENDIX A

### MOVEMENT DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for movement are documented in Appendix A, Movement Model Input Requirements, to Chapter 13 of the Period Processor (Section IV).

NOTES

II-14-A-2

## APPENDIX B

### MOVEMENT DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. Data used by the Movement Model are loaded by routine MOVELD, which in turn calls routine DPMOVE to provide a formatted listing of the data. Since data file 51 is used as input to MOVELD, these data cannot be loaded until after a satisfactory LDTOE run has been accomplished.

#### 2. ROUTINE MOVELD:

a. Purpose. This routine loads the constant data required by the Movement Model. The data are read from cards with identifications 0901, 0902, 0903, 1401, 1402, 1403, 1404, 1405, 1501, 1901, and 1902 and stored on data files 9, 14, 15, and 19.

b. Input Variables. The only input required by this routine is the data cards and the list of unit type designators from data file 51.

c. Output Variables. The output of this routine consists of the four loaded data files.

#### d. Logical Flow (Figure II-14-B-1):

(1) Block 1. The file name table array (IFNT) is brought into core and data files 9, 14, 15, and 19 are released, if they exist. Data file 9 is then initialized. If a data file 9 of the correct size (5 records of 1000 words) exists, data read serves to update the file; otherwise, the file will be allocated and set to zero.

(2) Block 2. The data file 15 parameters in IFNT are checked to determine if a file of the right size (16 records of 400 words) exists. If it does, data read that are normally stored on that file will update it. If the file does not exist, it will be allocated and set to zero initially.

(3) Block 3. A check is made to determine if data file 14 exists. If it does not, it is allocated and set to zero, and the variables IFOOD, MODE, MCPT, MAXRAT, NMCAT, LR14, and LR19 are given initial values. If data file 14 does exist, data read pertaining to the file will update it and the variables listed above will be initialized from its contents.

(4) Block 4. If data file 19 exists and has the correct number of records (50) and words (120) per record, data read will update it; otherwise, it is allocated and set to zero.

(5) Block 5. Data defining the travel mode mnemonics, mobility categories, and their legitimate combinations are read from cards identified by 1401 and stored on data file 14. The four-character travel mode mnemonics are stored in records 2 and 23 for the Blue and Red forces, respectively. The one-character mobility category codes are stored in words 3-7 of record

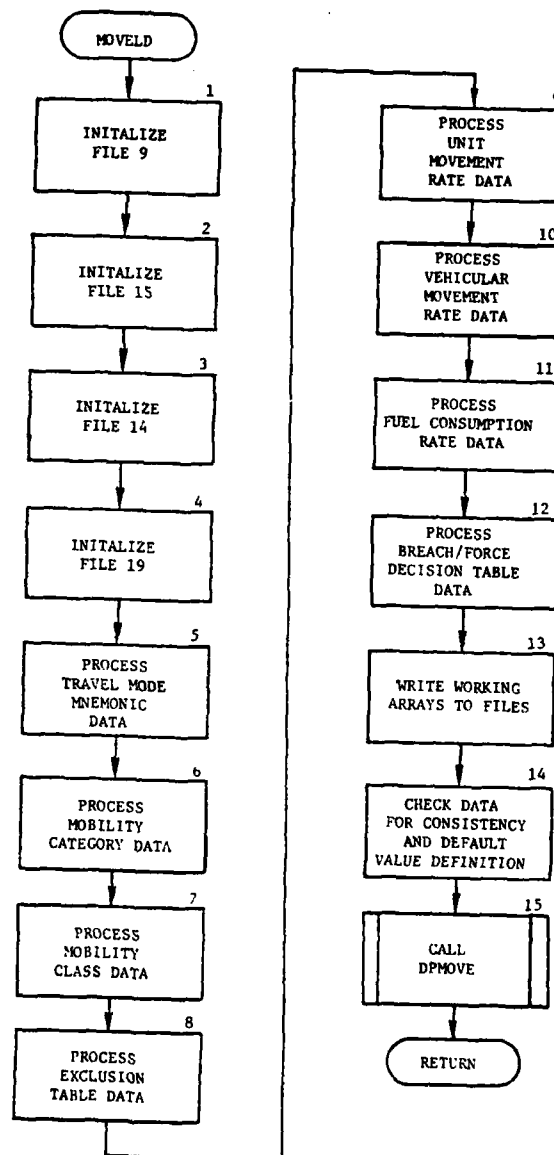


Figure II-14-B-1. Routine MOVELD

II-14-B-2

1 for the Blue force and words 8-12 for the Red force. The combination tables for the Blue force are stored in records 3-22 and those for the Red force are stored in 24-43. A maximum of 20 mobility categories and 20 travel mode mnemonics is allowed. As each mobility category is stored, the list is checked to assure that there are no duplicates. Each travel mode mnemonic is checked to assure it is legitimate.

(6) Block 6. The mobility category data are read from cards identified by 0901 and are stored on data file 9. The data are stored in the first record of data file 9 if they pertain to the Blue force and in the second record if they are Red force data. Each record is 1000 words long, allowing the mobility category associated with a particular UTD to be stored in the same location in the data file 9 record that the UTD appears in the 1000-word UTD list on data file 51. The card data associates a list of UTDs with a particular mobility category. An attempt is made to match each UTD with one in the list retrieved from data file 51. If a corresponding UTD is found, the data are stored as described above; otherwise, an error message is written.

(7) Block 7. The mobility class data are read from cards identified by 0902 and are also stored on data file 9. Words 211-410 and 411-610 of record 5 contain the Blue and Red mobility classes, respectively. The card data contain lists of item codes associated with a particular class. The item codes are used as pointers to store the class in the proper words of record 5. For example, if the Blue force card data indicated class 2 contained items 100 and 129, words 310 (210+100) and 339 (210+129) would be set to 2.

(8) Block 8. The mobility class exclusion table is read from card 1402 and is stored on data file 14 beginning in record 768. A pointer to the next available record in data file 14 is kept in data file 14 (record 1, word 17) and records may be added if they are needed. The exclusion tables contain a list of mobility classes that will be excluded from limiting the rate of movement of a type UTD unit moving by travel mode mnemonic TMDM. As each UTD is processed, it is assigned a record in data file 14 if one has not been assigned to it previously. This record number is stored in data file 9 (record 3, Blue; record 4, Red) in the word corresponding to the UTD's location in the UTD list. If the record is not newly assigned, it is searched for a word containing zero. The travel mode mnemonic and then the list of mobility classes to be excluded are stored in the record starting with that word. Each record is 20 words long. If the first 19 words cannot contain all of the mnemonics and classes, the next available record in data file 14 is assigned and its number is stored in word 20. One list of excluded classes with its associated UTD and TMDM is read per card.

(9) Block 9. The unit movement rate data are read from cards with the identifiers 1901 for road movement rate data and 1902 for cross-country movement rate data, and are stored in data file 19. Each record in data file 19 contains unit movement rates as a function of terrain, road, and movement type, and weather/light conditions. There is one record in data file 19 for each combination of unit formation and mobility category used. Two 3 by 20 word tables (information, category) containing pointers to records

in data file 19 are stored in data file 14 (records 44-49 for Blue force and 50-55 for Red force). The first of the two tables is associated with road movement rates and the second pertains to cross-country rates.

(10) Block 10. The cards identified by 1403 and 1404 contain vehicular movement rate data. The Blue force tables are stored on data file 14 in records 56-211, and the Red force tables are stored in records 212-367. The first 36 records of each group are associated with movement on roads and are functions of terrain type, road type, and weather/light conditions (2 by 3 by 6). The last 120 records contain cross-country movement rates as functions of terrain trafficability and weather/light conditions (20 by 6). The words of each record are associated with the mobility classes. Data contained on each card identify the mobility class and weather/light conditions that the movement rates are associated with.

(11) Block 11. Fuel consumption rate data (class III and class IIIA) are read from cards with identifiers 1501 and 1405. Data associated with vehicles moving on the ground (1501) are stored on data file 15. The 200 real words of each record correspond to the 200 possible items. The first eight records contain Blue force data and the last eight are used for Red force data. The consumption rates are functions of route type and activity. The aircraft consumption rate data (1405) are stored in data file 14 records 368-767, with the first 200 records of each group being used by the Blue force and the last 200 by the Red force. The 200 records per group correspond to the 200 equipment items. The first 16 words of each record are used to store the eight real values associated with aircraft fuel consumption. A table of item codes of aircraft for which fuel consumption data are loaded is kept in data file 9, record 5. Words 2-102 are reserved for Blue force data and words 104-204 for Red force. The first word of each group contains the number of item codes in the list.

(12) Block 12. The breach/force decision tables determine a unit's action if it encounters a barrier. They are read from cards identified by 0903 and are stored on data file 9, record 5. Words 663-831 are reserved for Blue force tables, and 832-1000 are used for Red. The first word of each group is set to the lowest of the movement priorities associated with the tables loaded for that force. The tables (a maximum of six is allowed) are stored in the remaining space. The tables contain four rows and seven columns. The first word of each row is the equipment loss cutoff value associated with that row. The other words contain decision indicators. Each data card contains a movement priority and the complete decision table associated with it. The priority determines the location at which the table is stored within the area assigned to the force.

(13) Block 13. The arrays used to accumulate data from cards are dumped to the appropriate files. Variables describing the files are written.

(14) Block 14. Checks are made to determine if items that the Movement Model will default to have been set. Checks are also made to assure that the various items of data are consistent.

(15) Block 15. Routine DPMOVE is called to print the data files just loaded.

3. ROUTINE DPMOVE:

a. Purpose. This routine dumps the constant data tables required by the Movement Model and loaded by MVELD from data files 9, 14, 15, and 19. The first part of the routine prints the data in tabular form with appropriate headers. The latter part gives a word dump of each file.

b. Input Variables. The only input to this routine is the data to be listed from data files 9, 14, 15, and 19, and the UTD list from data file 51.

c. Output Variables. The printed dumps described below are the output of this routine.

d. Processing Description. All Blue force data are listed, and then the Red force data are printed. The travel mode mnemonics, mobility category codes, and their combination tables are printed first for each force. Next are the mobility category codes and the UTDs associated with each code, followed by a list of the mobility classes and the item codes defining them. Next is a dump of the excluded mobility classes by UTD and travel mnemonic, followed by a listing of the unit road movement rate tables and the unit cross-country movement rate tables. Printed next are the vehicular road and cross-country movement rate tables, followed by tables of class III and class IIIA consumption rates. The tabular listings for the two forces are followed by a word dump of data file 14. Each record is dumped in integer format except records 368-767 which are dumped in real format. The records containing basic alphanumeric data are also dumped in alphanumeric format. Data file 9 is dumped next in integer format. The list of UTDs contained in data file 51 is also printed. Data file 15 containing class III consumption rate data is dumped in real format. Data file 19 is dumped in integer format and, finally a dump of the file name table is given.



NOTES

II-14-B-6

## APPENDIX C

### MOVEMENT DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. Ten different printouts are generated each time data are entered in the constant data file for movement. These printouts are listed in Figure II-14-C-1.

Printout Title	Data Source	Card ID
80-Column Card Image	Data Deck	All
Travel Mnemonic Mobility Category Index	File 14	1401
Mobility Category	File 9	0901
Mobility Class and Exclusion Table	File 9	0902
Unit Road Movement Rates	File 19	1901
Unit Cross Country Movement Rates	File 19	1902
Equipment Road Movement Rates	File 14	1403
Equipment Cross Country Movement Rates	File 14	1404
Ground Movement Fuel Consumption Rates	File 15	1501
Air Movement Fuel Consumption Rates	File 14	1405

Figure II-14-C-1. Movement Model Constant Data Input Printouts

2. EIGHTY-COLUMN CARD IMAGE. The 80-Column Card Image is printed so that the data entries are in the same format as entered in the card type. The format for the 80-Column Card Image is illustrated in Figure II-14-C-2. On the top line at the far left are the characters 1B, indicating card type 1 for Blue force. At the far right is the ID number 1401. Also on the top line are the characters ARAM and F indicating the army dismounted infantry unit, and the index cross referencing it to the mode of travel and its mobility category

is shown. For a complete description of the input card images, refer to Appendix A to Chapter 13 of the Period Processor for MOVELD input requirements.

3. MOBILITY CATEGORY AND TRAVEL MODE. The format for this report is illustrated in Figure II-14-C-3. The data and the titles of the columns are similar to the card format; a complete description may be gained by reference to Appendix A to Chapter 13 of the Period Processor for MOVELD input requirements, cards ID 1401 and 901.

4. MOBILITY CLASS AND EXCLUSION TABLE. For both Red and Blue forces the mobility class and exclusion tables are printed in the illustrated format of Figure II-14-C-4. The titles of the columns in this format are similar to those used in the punch card format; for a complete description, refer to Appendix A to Chapter 13 of the Period Processor for MOVELD input requirements for card ID 902 and 1402.

5. UNIT ROAD AND CROSS COUNTRY MOVEMENT. The rates at which a composite unit or basic unit type will move cross country and over the various roads are printed in this format. The format is illustrated in Figure II-14-C-5. The data from which this format is printed were taken from data file 19 and were originally punched into cards ID 1901 and ID 1902. The headings in the columns of this printout are the same as for the card formats. The upper half of this figure shows Blue force data for road movement, and the lower half shows the cross country movement rates. A similar printout is prepared for the Red force.

6. EQUIPMENT CROSS COUNTRY AND ROAD MOVEMENT RATES. The rates at which individual vehicles, wheeled and track-laying types, will move cross country and over the various roads are printed in this format. The format illustrated in Figure II-14-C-6 shows data for Blue force. The lower half shows road movement rates, and the upper half shows cross country movement rates. The data are taken from data file 14 and were originally punched in cards ID 1403 and ID 1404.

7. GROUND AND AIR FUEL CONSUMPTION RATES. The last printout in the constant data input printout series is for ground and air fuel consumption rates and is illustrated in Figure II-14-C-7. The upper portion of the figure shows Blue force ground equipment consumption rates and the lower portion shows the fuel consumption for blue force aircraft. Similar tables are printed for Red force. The data shown in the illustration were taken from data files 14 and 15 and were initially inscribed in cards ID 1405 and 1501.

# ACTIVITY RECORD DATA

-10	APR 1	FIVE FOOT	11	1401	1-
-10	APR 2	T FIVE FEET	111111	1401	2-
-10	APR 3	L FIVE FEET	111111	1401	3-
-10	APR 4	S FIVE FEET	111111	1401	4-
-10	APR 5	E FIVE FEET	111111	1401	5-
-10	APR 6			1401	6-
-10	APR 7			1401	7-
-10	APR 8			1401	8-
-10	APR 9			1401	9-
-10	APR 10			1401	10-
-10	APR 11			1401	11-
-10	APR 12			1401	12-
-10	APR 13			1401	13-
-10	APR 14			1401	14-
-10	APR 15			1401	15-
-10	APR 16			1401	16-
-10	APR 17			1401	17-
-10	APR 18			1401	18-
-10	APR 19			1401	19-
-10	APR 20			1401	20-
-10	APR 21			1401	21-
-10	APR 22			1401	22-
-10	APR 23			1401	23-
-10	APR 24			1401	24-
-10	APR 25			1401	25-
-10	APR 26			1401	26-
-10	APR 27			1401	27-
-10	APR 28			1401	28-
-10	APR 29			1401	29-
-10	APR 30			1401	30-
-10	APR 31			1401	31-
-10	APR 32			1401	32-
-10	APR 33			1401	33-
-10	APR 34			1401	34-
-10	APR 35			1401	35-
-10	APR 36			1401	36-
-10	APR 37			1401	37-
-10	APR 38			1401	38-
-10	APR 39			1401	39-
-10	APR 40			1401	40-
-10	APR 41			1401	41-
-10	APR 42			1401	42-
-10	APR 43			1401	43-
-10	APR 44			1401	44-
-10	APR 45			1401	45-
-10	APR 46			1401	46-
-10	APR 47			1401	47-
-10	APR 48			1401	48-
-10	APR 49			1401	49-
-10	APR 50			1401	50-
-10	APR 51			1401	51-
-10	APR 52			1401	52-
-10	APR 53			1401	53-
-10	APR 54			1401	54-
-10	APR 55			1401	55-
-10	APR 56			1401	56-
-10	APR 57			1401	57-
-10	APR 58			1401	58-
-10	APR 59			1401	59-
-10	APR 60			1401	60-
-10	APR 61			1401	61-
-10	APR 62			1401	62-
-10	APR 63			1401	63-
-10	APR 64			1401	64-
-10	APR 65			1401	65-
-10	APR 66			1401	66-
-10	APR 67			1401	67-
-10	APR 68			1401	68-
-10	APR 69			1401	69-
-10	APR 70			1401	70-
-10	APR 71			1401	71-
-10	APR 72			1401	72-
-10	APR 73			1401	73-
-10	APR 74			1401	74-
-10	APR 75			1401	75-
-10	APR 76			1401	76-
-10	APR 77			1401	77-
-10	APR 78			1401	78-
-10	APR 79			1401	79-
-10	APR 80			1401	80-
-10	APR 81			1401	81-
-10	APR 82			1401	82-
-10	APR 83			1401	83-
-10	APR 84			1401	84-
-10	APR 85			1401	85-
-10	APR 86			1401	86-
-10	APR 87			1401	87-
-10	APR 88			1401	88-
-10	APR 89			1401	89-
-10	APR 90			1401	90-
-10	APR 91			1401	91-
-10	APR 92			1401	92-
-10	APR 93			1401	93-
-10	APR 94			1401	94-
-10	APR 95			1401	95-
-10	APR 96			1401	96-
-10	APR 97			1401	97-
-10	APR 98			1401	98-
-10	APR 99			1401	99-
-10	APR 100			1401	100-

Figure II-14-C-2. Movement Model 80-Column Card Image

FOFC73LUF

MOBILITY CATEGORY CODE	UNIT TYPE DESIGNATORS
F	2 AY
T	KMCT QNFA KAMI KLSI KTCT KMTY KCST ITCT ICST PMFA PMNT PMCT IMCT ILSI QNFA QTSI PIMI PICT IMHC EANT ITMT EQMI IAMI
L	IMCC QPCC PTCC JAGA TCCT JCLY IPCT JOLI TACC JALC ILLY ILLI TSLY IJLY IAPH VALH VCLH ITCH TRLH JACH JCSH VFLH VBLH IFCA TFLA IGCA TGLA KCCA KSLA KGLA KALA KCLA KCCA KCLA KCHA KLLA JHCH JFCF IHCC IHLC IJCH JHCL JLLI JSLC KSLC JTLH PTLH JFLL KACH KACH KALH KOLE KLEL KMCT ICTY TALH FCMH EQSH JCLH QCCC FTIJ FSLI FNLL FALL FNLL
C	IKAH IRMH JASH JSSH TFFA IRFA IGFA KGFA KHFA KQFA KLFA JAFD JQFD RASF QCCF RSCF QSCF QSCF FSCS JSCS PQCS IA4H FWHH PGMH KHFH KASH KQSH RQSF QCCF EAPH EFAF EGFA IGFA FGFA FHFA FSFA FMFA EASD EFSE JCSE ICSE FSSS EQMC FWHF EASH FESE KCSE KCSH
F	PRSC QCSF PTHF QHNF JQSE KPSF
T	IACT IPMT IFSI IPCI ICMI IMST EOMI EQMI

ГОРЬКОГЪЗНАНІЯ

[illegible]

**Figure II-14-C-3. Mobility Category and Travel Mode**

II-14-C-4

FORCEX' HF

# MOBILITY CLASS TABLE (EQUIPMENT ITEM)

MOBILITY CLASS	EQUIPMENT ITEM CODES
2	74 160
3	61 161 162 164 165 166 167
4	135 138 163
5	25 29 35 41 71 73 75 185 186 188
6	37 52 56 58 187 189 190 191
7	140 141
8	85 86 87 88

FORCEX' BLUE

## MOBILITY CLASS EXCLUSION TABLE

UTD	TRAVEL MONITOR	MOBILITY CLASSES TO BE EXCLUDED
ITCT	TCCM	2 3 4
ITCT	TCCM	2 3 4
	TCCM	2 3 4
	TCCM	2 3 4
PMFA	TCCM	2 3 4
	TCCM	2 3 4
	TCCM	2 3 4
TMCT	TCCM	2 3 4
	TCCM	2 3 4
	TCCM	2 3 4
TLST	TCCM	2 3 4
	TCCM	2 3 4
	TCCM	2 3 4
PMFA	TCCM	2 3 4
	TCCM	2 3 4
	TCCM	2 3 4
PTST	TCCM	2 3 4
	TCCM	2 3 4
	TCCM	2 3 4

Figure II-14-C-4. Mobility Class and Exclusion Table

II-14-C-5

# MOBILITY CATEGORY ROAD MOVEMENT RATE TABLES (KM/HR)

TABLE 11-14-C-5

MOBILITY CATEGORY CODE	MOBILITY FORMATION	SOURCE TYPE	ADMINISTRATIVE MOVE			TACTICAL MOVE		
			ROAD TYPE			ROAD TYPE		
			T1	T2	T3	T1	T2	T3
T	"	DAY - NO RAIN	0	0	0	0	0	0
		DAY - LIGHT RAIN	0	0	0	0	0	0
		DAY - HEAVY RAIN OR FOG	0	0	0	0	0	0
		NIGHT - NO RAIN	0	0	0	0	0	0
		NIGHT - LIGHT RAIN	0	0	0	0	0	0
L	"	DAY - NO RAIN	40	36	40	36	40	36
		DAY - LIGHT RAIN	40	36	40	36	40	36
		DAY - HEAVY RAIN OR FOG	32	28	32	28	32	28
		NIGHT - NO RAIN	36	32	36	32	36	32
		NIGHT - LIGHT RAIN	32	28	32	28	32	28

# MOBILITY CATEGORY CROSS COUNTRY MOVEMENT RATE TABLES (KM/HR)

TABLE 11-14-C-6

MOBILITY CATEGORY CODE	MOBILITY FORMATION	TRAFFICABILITY INDEX											
		1	2	3	4	5	6	7	8	9	10	11	12
		1	2	3	4	5	6	7	8	9	10	11	12
T	"	DAY - NO RAIN	3	3	3	3	3	3	3	3	3	3	3
		DAY - LIGHT RAIN	3	3	3	3	3	3	3	3	3	3	3
		DAY - HEAVY RAIN OR FOG	3	3	3	3	3	3	3	3	3	3	3
		NIGHT - NO RAIN	2	2	2	2	2	2	2	2	2	2	2
		NIGHT - LIGHT RAIN	2	2	2	2	2	2	2	2	2	2	2
L	"	DAY - NO RAIN	0	0	0	0	0	0	0	0	0	0	0
		DAY - LIGHT RAIN	0	0	0	0	0	0	0	0	0	0	0
		DAY - HEAVY RAIN OR FOG	0	0	0	0	0	0	0	0	0	0	0
		NIGHT - NO RAIN	0	0	0	0	0	0	0	0	0	0	0
		NIGHT - LIGHT RAIN	0	0	0	0	0	0	0	0	0	0	0

Figure 11-14-C-5. Unit Road and Cross Country Movement Rates

FOR REFERENCE

MOBILITY CLASS

MOBILITY CLASS

MOBILITY CLASS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
DAY - NO RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
DAY - LIGHT RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
DAY - HEAVY RAIN OR FOG	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
NIGHT - NO RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
NIGHT - LIGHT RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
NIGHT - HEAVY RAIN OR FOG	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

II-14-C-7

FOR REFERENCE

MOBILITY CLASS ROAD MOVEMENT RATE TABLES (K4/42)

MOBILITY CLASS	ROUTE TYPE - PA	ROUTE TYPE - T1	ROUTE TYPE - T2	ROUTE TYPE - T3	ROUTE TYPE - T4	ROUTE TYPE - T5	ROUTE TYPE - T6	ROUTE TYPE - T7	ROUTE TYPE - T8	ROUTE TYPE - T9	ROUTE TYPE - T10	ROUTE TYPE - T11	ROUTE TYPE - T12	ROUTE TYPE - T13	ROUTE TYPE - T14	ROUTE TYPE - T15	ROUTE TYPE - T16	ROUTE TYPE - T17	ROUTE TYPE - T18	ROUTE TYPE - T19	ROUTE TYPE - T20
1	DAY - NO RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	DAY - LIGHT RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	DAY - HEAVY RAIN OR FOG	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	NIGHT - NO RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	NIGHT - LIGHT RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	NIGHT - HEAVY RAIN OR FOG	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	DAY - NO RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	DAY - LIGHT RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	DAY - HEAVY RAIN OR FOG	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	NIGHT - NO RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	NIGHT - LIGHT RAIN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	NIGHT - HEAVY RAIN OR FOG	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

Figure II-14-C-6. Equipment Road and Cross Country Movement Rates



# CONSUMPTION RATES OF CLASS III SUPPLIES

FORCEBLUE

EQUIPMENT ITEM CODE	UNIT- ACTIVITY- CODE	GALS/HR	WAVE	TG	TD	TEMPERATURE- LE		STAY	
						90/32	90/32	LE	STAY
1		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
8		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
9		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
25		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
26		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
27		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
28		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
29		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

# CONSUMPTION RATES OF CLASS IIIA SUPPLIES (GALS/HR)

FORCEBLUE

EQUIPMENT ITEM CODE	ALTITUDE CUT-OFF (FEET)	CRUISE SPEED (KNOTS)	CONSUMPTION RATES				ACTIVITY- TEMP.-		STAY	
			LE SPEED LF ALT.	LE SPEED GT ALT.	GT SPEED LF ALT.	GT SPEED GT ALT.	TEMP.-	LE	STAY	GT
45	0.0	0.0	0.00	0.00	0.00	22.00		0.00	0.00	0.00
46	0.0	0.0	0.00	0.00	0.00	85.00		0.00	0.00	0.00
47	0.0	0.0	0.00	0.00	0.00	467.00		0.00	0.00	0.00
48	0.0	0.0	0.00	0.00	0.00	43.50		0.00	0.00	0.00
49	0.0	0.0	0.00	0.00	0.00	170.00		0.00	0.00	0.00

Figure II-14-C-7. Ground and Air Fuel Consumption Rates

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR MOVEMENT

(AVAILABLE UNDER SEPARATE COVER)

II-14-D-1

NOTES

II-14-D-2

## CHAPTER 15

### ENGINEER DATA INPUT PROGRAMS

1. INTRODUCTION. The Engineer data input programs consist of two independently executed routines, ENGLD and BARLD. These routines create and load the files required by the Engineer Model and other models using barrier information. The data loaded define the time and resources required to accomplish each engineering task and describe the location and characteristics of the barriers.

#### 2. ROUTINES:

a. General. The following routines comprise the Engineer data input programs.

b. Routine ENGLD. This routine reads and edits the data cards describing an engineering task and loads the information on the file. It also calls DUMP17.

c. Routine DUMP17. This routine provides a formatted dump of the Engineer data file.

d. Routines GET17, IFILE, IDENT, and ITYPCK. These four routines are efficiency routines used by ENGLD (GET17 is also used by DUMP17). GET17 is responsible for the input/output operation of bringing in the proper record from data file 17. IFILE edits the file number associated with each data card for the file, IDENT edits the card number, and ITYPCK edits the card type.

e. Routine BARLD. This routine reads and edits the input data describing the barriers and loads it on the file. It also creates the eight-digit quadrature code associated with each barrier. Finally, it provides a formatted dump of the data loaded.

#### 3. FILES. The following files are used as indicated:

- . Barrier description - data file 2
- . Engineer task data - data file 17

NOTES

II-15-2

## APPENDIX A

### ENGINEER DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for engineer activities are documented in Appendix A, Engineer Model Input Requirements, to Chapter 14 of the Period Processor (Section IV).

NOTES

II-15-A-2

## APPENDIX B

### ENGINEER DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. Engineer data are loaded by the routines described in this appendix. The routine ENGLD loads the constant data describing the various engineer tasks and, through a call to routine DUMP17, provides a listing of the data. Routine BARLD loads and lists data describing actual barriers. ENGLD and BARLD are run as independent jobs.

#### 2. ROUTINES ENGLD AND DUMP17:

a. Purpose. Routine ENGLD is responsible for editing and loading the data for engineer tasks onto data file 17. DUMP17 prints the contents of data file 17 after it has been loaded. ENGLD reads a data card, prints its image, and examines this card for possible loading errors. Should an error occur, this routine identifies the type of error, prints the diagnostic message, places useable information onto the file, and reads the next card. Data file 17 contains data pertaining to actual engineering task information; and as such, six different card types are permissible, each providing different types of data for tasks. After reading the data, and placing it on data file 17, ENGLD calls routine DUMP17 to print those records containing data.

#### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Description</u>
IDATA	DF17	Record number 157 for Red force.

#### c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Description</u>
IDATA	DF17	Data to be loaded.
RDATA	DF17	Rate information, minimum and standard.

#### d. Logical Flow (Figure II-15-B-1):

- (1) Block L1. Initialize variables and zero output array.
- (2) Blocks 1, L10001, and 2. Bring in and print the file name table and create data file 17 on disk.
- (3) Block 3. Zero data file 17.
- (4) Block L10. Read next card.
- (5) Blocks 4, L15, L20, L25, L30, 5, and L1100. Routines IFILE, IDENT, and ITPCHK edit the first data card for proper file, card, and card



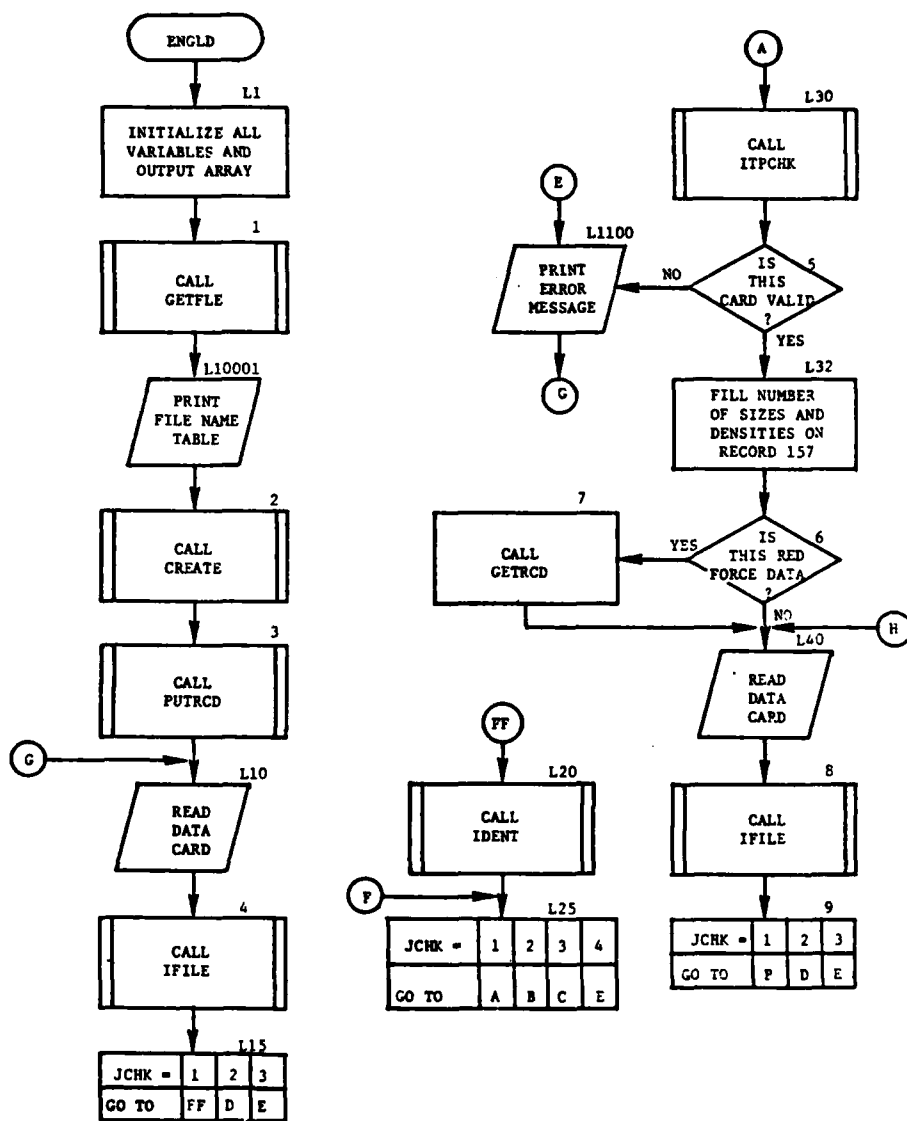


Figure II-15-B-1. Routine ENGLD (Continued on Next Page)

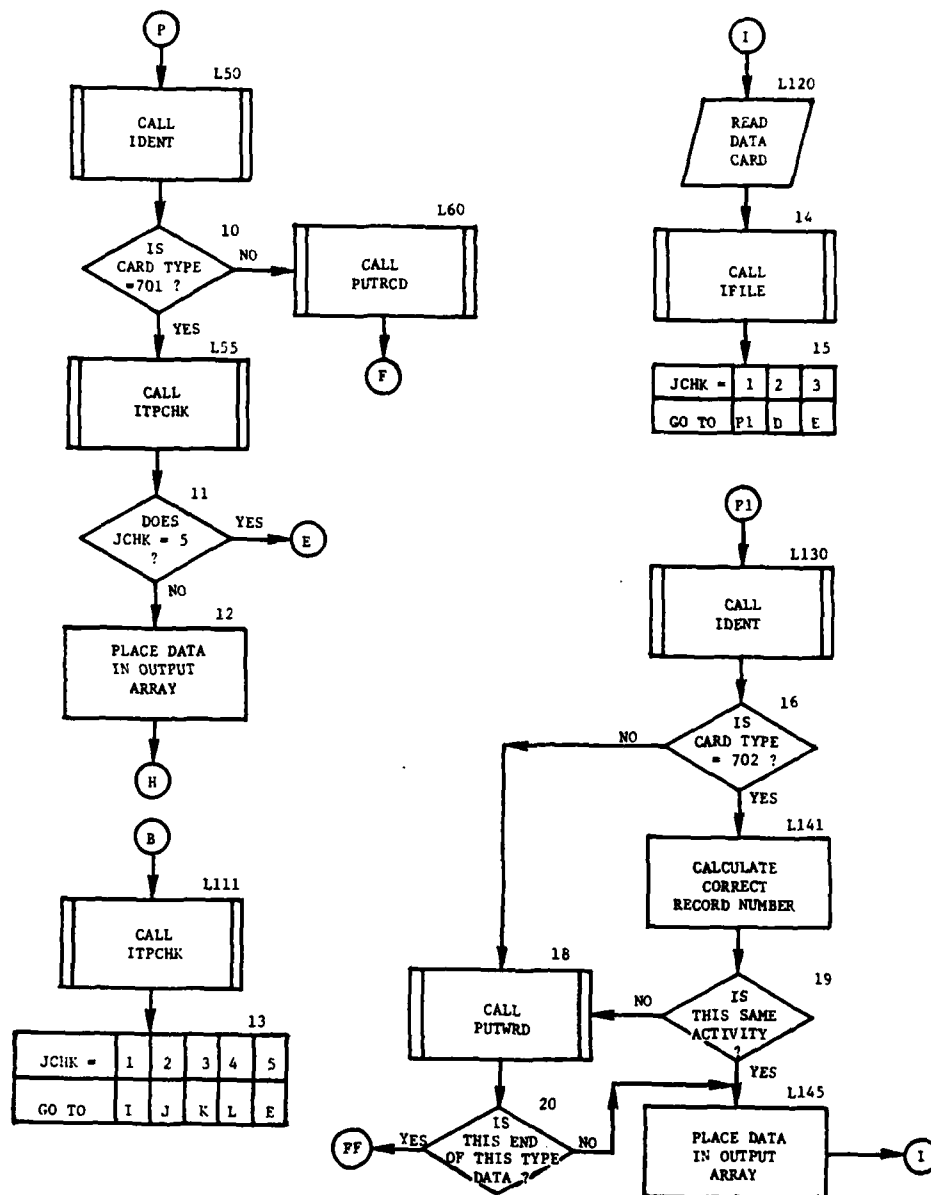


Figure II-15-B-1. Routine ENGLD (Continued)

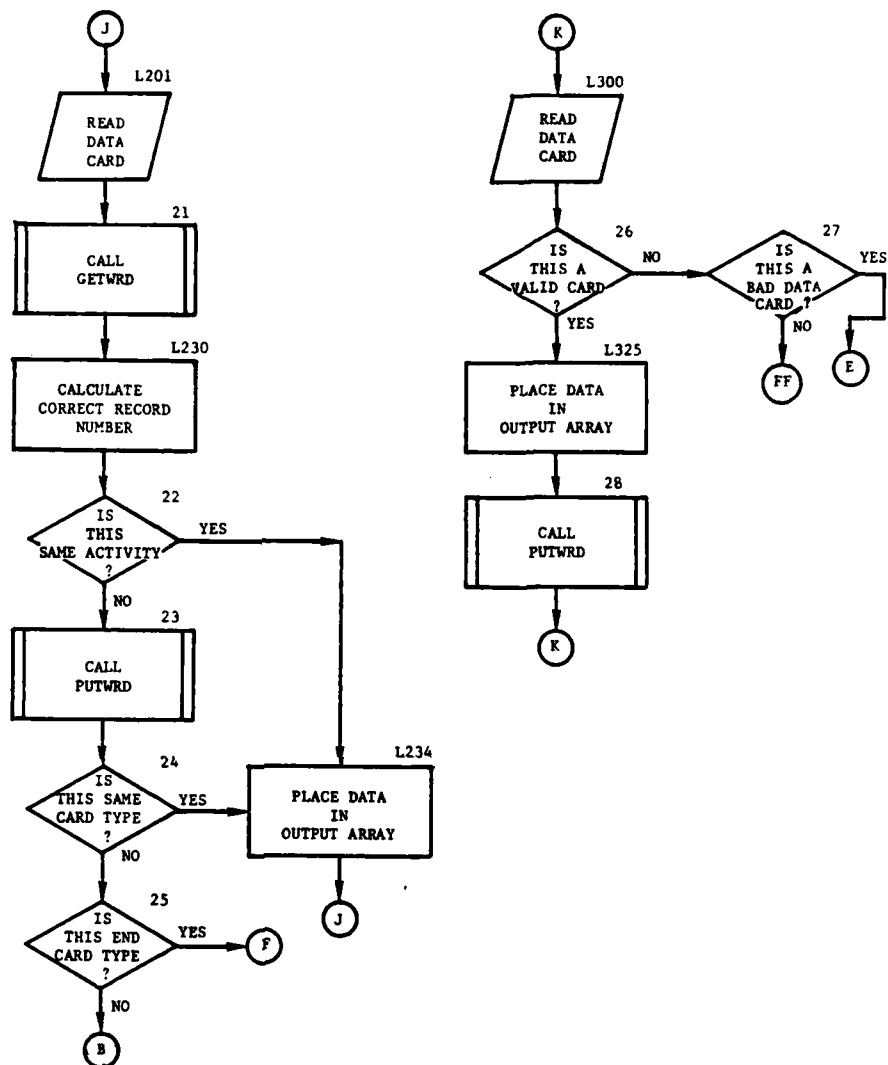


Figure II-15-B-1. Routine ENGLD (Continued)

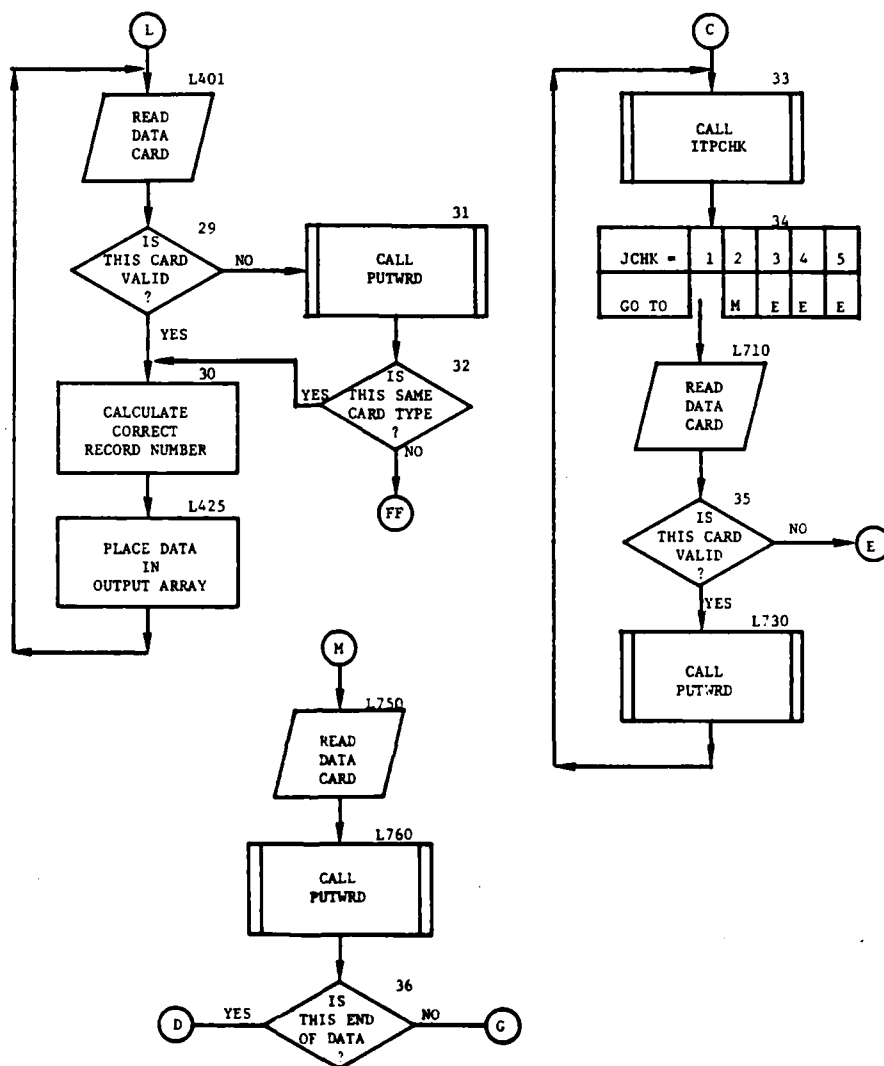


Figure II-15-B-1. Routine ENGLD (Continued)

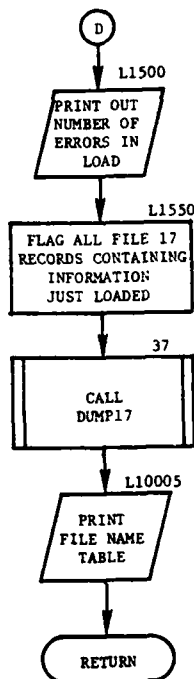


Figure II-15-B-1. Routine ENGLD (Concluded)

type; control branches on card type or end of data. If this is not a valid card, the diagnostic message is printed and control goes to block L10 to read the next data card.

(6) Blocks L32, 6, and 7. Fill in possible number of sizes and densities by default for all task types. If this is a Red data card, bring in record 157.

(7) Blocks L40, 8, 9, L50, 10, L55, 11, 12, and L60. Read next card of first card type, containing task type, description, basic task size, number of sizes being played, and number of densities and restricted movement rate; all are placed on record 157. If this is not the end of the data, and there were no errors detected on this card, processing continues. If this is still the first card type, place the information just read into the output array. If this is a different card type, put record 157 onto the disk, and transfer control to block L25.

(8) Blocks L111 and 13. Card 1702 has four possible types; control branches on the proper type.

(9) Block L120. Read task type, activity, item code, minimum amount needed, standard amount required, proportionality factor, transportation item code, and modifiers affecting the task rates if less than the standard amount needed.

(10) Blocks 14, 15, L130, 16, L141, 18, 19, 20, and L145. If this is a valid card, and not the end of the data, processing continues. If this is the same card type, put the information just read into the output array. If this is a different card type, place the output array onto data file 17, and branch control to block L120.

(11) Blocks L201, 21, and L230. Read force pertaining to data, task type, size step, density, minimum troop level required for this size task, rate associated with this troop level, standard troop level, and standard rate. Bring in part of record 157 and calculate the record number where data are to be stored.

(12) Blocks 22, 23, 24, 25, and L234. If no errors are detected from this card, and this does not end the data, continue processing. If this is the same card type, put the data into the output array. If not, put the output array on the proper record of data file 17 and transfer control to block L201.

(13) Block L300. Read task type, activity, nighttime rate modifier, and terrain modifiers affecting the task rate on this type terrain.

(14) Blocks 26, 27, L325, and 28. If this is a valid card, and not the end of the data, processing continues. Put this data into output array and place it on data file 17; transfer control to block L300.

(15) Blocks L401, 29, 30, 31, 32, and L425. Read task type, activity, size step, density class, force rate, forcing code, and associated casualties.

If this is a valid card, and not the end of the data, processing continues. Put these data into the output array and place it on data file 17; transfer control to block L401.

(16) Blocks 33, 34, L710, 35, and L730. Read the item code and the contingency associated with it.

(17) Blocks L750, L760, and 36. Read contingency associated with each troop type, and place this onto data file 17, record 157. If this is not the end of the data, control goes to block L10.

(18) Blocks L1500, L1550, 37, and L10005. When the last data card has been read, print the total number of errors found. The last word on every record containing valid data is marked, and DUMP17 is called. The file name table is printed and control returns to the calling routine.

3. ROUTINES GET17, IFILE, IDENT, AND ITYPCK. These four routines are efficiency routines used by ENGLD (GET17 is also used by DUMP17). GET17 is responsible for the input/output operation of bringing in the proper record from data file 17. IFILE edits the file number associated with each data card for the file, and returns a three if an error is encountered. IDENT edits the card number returning a four if an error has occurred, and ITYPCK edits the card type and returns a five if an error occurs; otherwise, IDENT and ITYPCK return the values found for card number and card type, respectively.

#### 4. ROUTINE BARLD:

a. Purpose. BARLD places the barriers to be played in any period or game onto data file 2 of the constant data base, by either loading a new barrier, updating an existing barrier, or removing a barrier. It also creates the eight-digit quadrature code associated with each barrier segment and places it onto data file 22. The routine prints the barrier segments line-by-line.

##### b. Input Variables:

(1) Barriers used in the actual barrier line segment are those common to the entire Engineer Model.

##### (2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
XMAX	DF37	Maximum X coordinate permissible.
YMAX	DF37	Maximum Y coordinate permissible.
XACTON	DF37	X coordinate approximating the center of action.

<u>Name</u>	<u>Source</u>	<u>Contents</u>
YACTON	DF37	Y coordinate approximating the center of action.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
KDUM(1)	DF22	Link for this code to corresponding barrier record.
KDUM(2)	DF22	Eight-character quadrature code.

d. Logical Flow (Figure II-15-B-2):

- (1) Block 1. Bring in the file name table and print it.
- (2) Blocks 2 and L1000. Read geometry necessary for quadrature and print.
- (3) Blocks 3 and 4. Read mnemonic of barrier line and card type; if card is not a load type, transfer control to block L2000.
- (4) Blocks 5, 6, 7, 8, and 9. Remove, create, and zero data files 2, 18, 22, and 37. Place geometry information on data file 37.
- (5) Block L10. Read mnemonic of next barrier line and card type.
- (6) Blocks 10 and 11. Read barrier segment mnemonic; both barrier end points; Blue/Red task, size, troop types; and the segment's intelligence and physical status. Check for end of barrier line; if end of barrier line is found, transfer control to block L10.
- (7) Blocks L11 and 12. Call routine BUIDRC to obtain the correct identifying record number for this barrier. Call routine CRTQD to create the eight-digit quadrature code based on the location of this barrier segment.
- (8) Blocks 13, 14, and 15. If the card type is not UPDT, the record number of the present segment is put into the record for the preceeding segment, and the barrier information is put in data file 2. If the card type is UPDT, the record number is retained.
- (9) Blocks L3001 and L3005. The barrier file and the intelligence and existence status of the barrier are printed.
- (10) Blocks 16 and 17. If the card is a UPDT type, transfer control to block L10 to read the next header card. If not, the record number and quadrature code are put into data file 22.
- (11) Block L100. This block is the end of the loop that reads barrier segment data. If end of barrier line, transfer control to block L10; otherwise, transfer control to block 10.



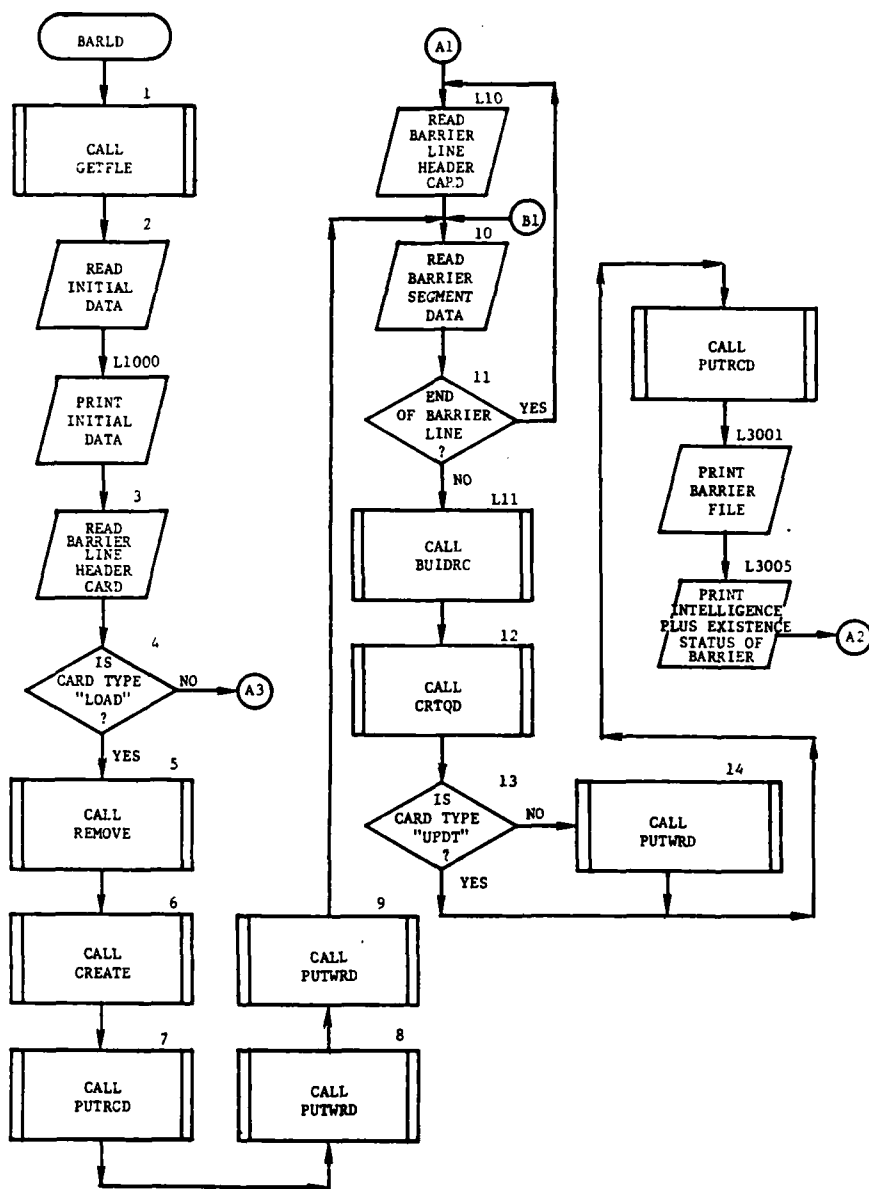


Figure II-15-B-2. Routine BARLD (Continued on Next Page)

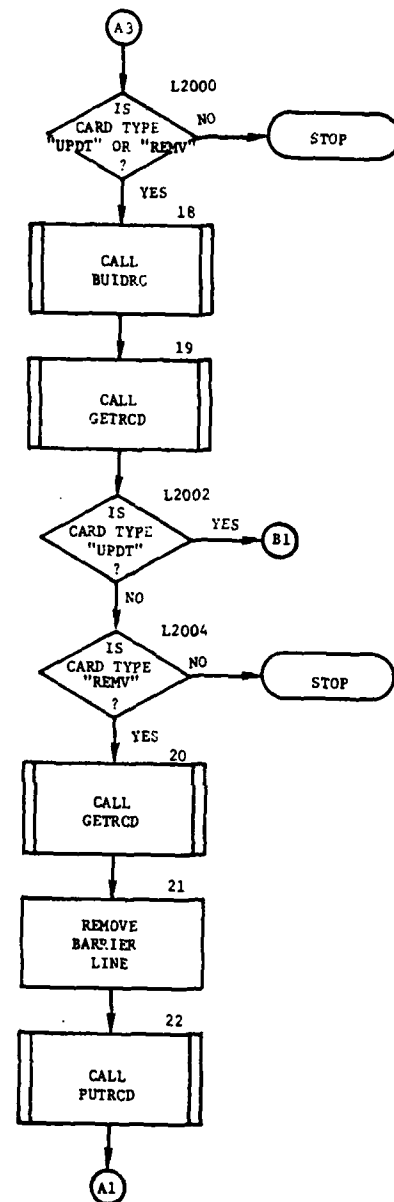
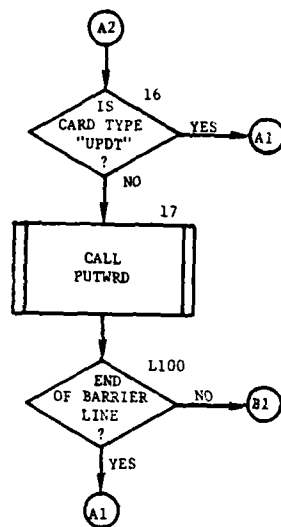


Figure II-15-B-2. Routine BARLD (Concluded)

(12) Blocks L2000, 18, and 19. If card type is not UPDT or REMV, stop the program; otherwise, determine the record number and bring it from data file 2.

(13) Blocks L2002 and 2004. If the card type is UPDT, transfer control to block 10 to read the next data card. If not, and if the card type is not REMV, stop the program.

(14) Blocks 20, 21, and 22. If the card type is REMV, remove the barrier line from data file 22, and transfer control to block L10.

## APPENDIX C

### ENGINEER DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. This appendix contains sample printouts of constant data input for Combat Service Support. The input of barrier and facility file data results in the generation of one type of output report and the input of engineer task file data results in the generation of five types of output reports.
2. BARRIER SEGMENT LOCATION AND STATUS. The format of this printout is illustrated in Figure II-15-C-1. The first line gives the coordinates from card ID 0200, type 1 (geographical area and battle area); this appears only once. The second line is the barrier record number and the quadrature number used for search purposes. The third line is the barrier line name from card ID 0201, type 1. The first two columns of the fifth line are provided to give record numbers of previous and following segments of the same barrier line; the remaining data are transferred directly from appropriate cards. More barrier segments follow in the same format.
3. ENGINEER 80-COLUMN CARD IMAGE. This format is illustrated in Figure II-15-C-2. The sources of this printout are the original cards that brought the data into the model. The number in the left column is the card type number and the number in columns 73-76 is the card ID.
4. ENGINEERING TASK PARAMETERS. This format, illustrated in Figure II-15-C-3, has three parts. The first part, task sizes, is a printout of data from card ID 1701, type 1; the second part, contingency levels of equipment, lists data from card ID 1703, type 1; and the third part, contingency levels of troops, lists data from card ID 1703, type 2.
5. ENVIRONMENTAL AND TROOP RATES AND RATE MODIFIERS. This format, illustrated in Figure II-15-C-4, has three parts. The first part, environmental rate modifiers, is a printout of data from card ID 1702, type 3; the second part, minimum number of troops and associated rates, and the third part, standard number of troops and associated rates, are printouts of data from card ID 1702, type 2.
6. EQUIPMENT REQUIREMENTS AND RATE MODIFIERS. This format, illustrated in Figure II-15-C-5, has two parts, both of which are printouts of data from card ID 1702, type 1. The upper part covers minimum and standard equipment requirements, and the lower part the associated rate modifiers.
7. MINEFIELD FORCING CASUALTIES. This format, illustrated in Figure II-15-C-6, is a printout of equipment types and their losses when forcing a minefield. It is a printout of data from card ID 1702, type 1.

21503	17700	16298	128000
-------	-------	-------	--------

ITOT/L = 3376ALLOC = 4237000  
PARFIER LINE ALPHA 31

CPNMC FLNCD	XFC01	YCOPD	YCOPC	YCORZ	BUTD	SIZEB.	CAPIC	IDNTB	IDMTR	ISIZER
0	0	168000.0	153000.0	148600.0	151000.0	RIV001	0	00000000	0000	0
		INTELLIGENCE STATUS	BOTH	---	---	EXISTENCE STATUS	ALL			

ITOTAL = 601ALLOC = 42333450  
PARPIER LINE ALPHA 21

PNMCC	FLNMCD	XCCPF1	YC0R01	XC0P02	YC0Q02	RUID	SI7F8	CHARC	IDNFB	IONTR	ISIZER
3276	0	1446C1.0	151C06.0	149C06.0	15081C.0	BFXJ01	EXISTENCE	STATUS	ALL	0124	0124
			INTELLIGENCE	STATUS	NOTM						20

IYOTAL = 3377ALLOC = 4233376C  
 PARFIER LINE ALPHA 31

CPYMMCD	FLNMCD	YCORE1	YCORD1	YCORD2	YCORD2	RLID	ST7EB	CHARIC	IONTB	IONTP	ISIZER
EC1	0	1490J1.0	150830.0	150700.0	150600.0	RTV002		0	LCCLC32C	0000	0000
			INTELLIGENCE STATUS	BOTH	EXISTENCE STATUS	-ALL					

ITOTAL = 3376ALLOC = 73206900  
 PARFIER LINE ALPHA 91

PUNMC FLNMC0	XCORF1	YCOR01	YCOR02	YCOR02	BUID	SIZEB	CT4P1C	10M1B	10M1R	151Z1F
3777	0	150700.0	150600.0	150900.0	RYW03	15000.0	ALL	000032.	0000	0
		INTELLIGENCE STATUS	INTELLIGENCE STATUS	INTELLIGENCE STATUS	EXISTENCE STATUS	EXISTENCE STATUS	ALL	000032.	0000	0

**Figure II-15-C-1. Barrier Segment Location and Status**

1R 2	MINIFIELD AT-AD	3050	1	320	17 1	1
1R 5	BRIDGE FLOATS MAR	3050	1	400	17 1	2
1R 6	BRIDGE FLOATS MATK	3050	1	400	17 1	3
1R 7	BRIDGE FIXED AVLD	1800	1	400	17 1	4
1R 8	RAFT/FERRY MAP	5060	1	5	17 1	5
1R 10	COGN	5060	1	240	17 1	6
1R 11	BRIDGE LOCAL 1-LANE	5060	1	400	17 1	7
1R 12	BRIDGE LOCAL 2-LANE	5060	1	900	17 1	8
1R 13	BRIDGE LOCAL 3-LANE	5060	1	1200	17 1	9
1R 14	BRIDGE LOCAL 4-LANE	5060	1	1500	17 1	10
1R 15	CRATER ADM	10060	1	-0	17 1	11
1R 16	TRF PLONDDUN	5060	1	320	17 1	12
1R 20		-0-0-0		-0	17 2	-0
1R 3	BUILD 1	X130161		-0 -0	10017 2	1
1R 3	BUILD 1015000	X 13161		-0 -0	10017 2	2
1R 3	BUILD -0	X -0 -0		-0 -0	-0 -017 2	3
1R 3	BEACH 5	X 2160		-0 -0	10017 2	4
1R 3	BEACH160	X250160		-0 -0	10017 2	-0
1R 3	BEACH -0	X -0 -0		-0 -0	-0 -017 2	5
1R 3	BEACH 5 1000	X 13161		-0 -0	20017 2	6
1R 3	BEACH161	X250161		-0 -0	10017 2	-0
1R 3	BEACH -0	X -0 -0		-0 -0	-0 -017 2	7
1R 5	BUILD 10	X130175		-0 -0	10017 2	8
1R 5	BUILD -0	X -0 -0		-0 -0	-0 -017 2	9
1R 5	BUILD137	X130161		-0 -0	150017 2	12
1R 6	BUILD135	X250163		-0 -0	410017 2	13
1R 6	BUILD163	X250163		-0 -0	210017 2	14
1R 6	BUILD161	X250161		-0 -0	150017 2	15
1R 6	BUILD -0	X -0 -0		-0 -0	-0 -017 2	16
1R 6	BEACH 5	X 2160		-0 -0	50010017 2	17
1R 6	BEACH160	X250160		-0 -0	110017 2	-0
1R 6	BEACH -0	X -0 -0		-0 -0	-0 -017 2	18
1R 6	BEACH135	X130161		-0 -0	210017 2	19
1R 6	BEACH161	X250161		-0 -0	150017 2	20
1R 6	BEACH163	X250163		-0 -0	110017 2	-0
1R 6	BEACH -0	X -0 -0		-0 -0	-0 -017 2	21

Figure II-15-C-2. Engineer 80-Column Card Image

# ENGINEERING TASK PARAMETERS

TASK TYPE	BASIC STRE	NO. STRE	NO. DECESS. INCR.	CONSTRUCTIVE RATE
1	3	50	1	0
2	100	50	1	333
3	333	50	1	-500
4	0	50	1	0
5	0	50	1	400
25	0	50	1	0
	0	50	1	0

BLUE FORCE			RED FORCE		
EQUIP	CONT. LEVEL		EQUIP	CONT. LEVEL	
1	0		1	0	
2	0		2	0	
3	0		3	0	
4	0				
5	0		5	0	
199	0		199	0	
200	0		200	0	

UNIT	CONT. LEVEL	UNIT	CONT. LEVEL
MNVR	0	MNVR	0
ENGR	0	ENGR	0
ADM	0	ADM	0
FLBR	0	FLBR	0
MAR	0	MAR	0

II-15-C-4

Figure II-15-C-3. Engineering Task Parameters

FORCE TASK		ACTIVITY		LIGHT/DARKNESS		MODIFIERS			
						A	B	C	D F
PLUE	3	BUILD		50		-0	-0	-0	-0 -0
MINIMUM NUMBER OF TROOPS AND THE ASSOCIATED RATE									
SIZE	TROOPS	RATE	TROOPS	RATE	TROOPS	RATE	TROOPS	RATE	TROOPS
1	1	50.000							
2	1	50.000							
3	1	50.000							
4	1	50.000							
5	1	50.000							
6	0	0							
STANDARD NUMBER OF TROOPS AND THE ASSOCIATED RATE									
SIZE	TROOPS	RATE	TROOPS	RATE	TROOPS	RATE	TROOPS	RATE	TROOPS
1	1	50.000							
2	2	50.000							
3	3	50.000							
4	4	50.000							
5	5	50.000							
6	0	0							
7	0	0							
8	0	0							
9	0	0							
10	0	0							
11	0	0							

Figure II-15-C-4. Environmental and Troop Rates and Rate Modifiers



ITEM CODE	ITEM CODE	EQUIPMENT			EQUIPMENT			EQUIPMENT			EQUIPMENT			VEHICLE
		MIN.	MOD.	MIN.	MIN.	MOD.	MIN.	MIN.	MOD.	MIN.	MIN.	MOD.	MIN.	
161	161	0	0	0	0	0	0	0	0	0	0	0	0	161
10	10	0	0	0	0	0	0	0	0	0	0	0	0	161

ITEM CODE	ITEM CODE	EQUIPMENT			EQUIPMENT			EQUIPMENT			EQUIPMENT			VEHICLE
		MIN.	MOD.	MIN.	MIN.	MOD.	MIN.	MIN.	MOD.	MIN.	MIN.	MOD.	MIN.	
161	161	0	0	0	0	0	0	0	0	0	0	0	0	161
10	10	0	0	0	0	0	0	0	0	0	0	0	0	161

Figure II-15-C-5. Equipment Requirements and Rate Modifiers



NOTES

II-15-C-8

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR ENGINEER

(AVAILABLE UNDER SEPARATE COVER)

II-15-D-1

NOTES

II-15-D-2

## CHAPTER 16

### AIRMOBILE DATA INPUT PROGRAMS

1. INTRODUCTION. The Airmobile data input program creates the necessary file and loads all data used by the Airmobile Model including data required to define the composition of a mission and the weapons and aircraft characteristics necessary to simulate ground to air attrition.
2. ROUTINES. The Airmobile data input program is the routine FIL7LD. This routine is responsible for reading and editing the data cards containing the Airmobile data and loading the extracted data on the file. It also gives a partial formatted dump of the file and a complete unformatted record dump of the file.
3. FILES. Data file 7 is used to contain the airmobile data.

NOTES

II-16-2

## APPENDIX A

### AIRMOBILE DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for airmobile events are documented in Appendix A, Airmobile Model Input Requirements, to Chapter 15 of the Period Processor (Section IV).



NOTES

II-16-A-2

## APPENDIX B

### AIRMOBILE DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. The constant data stored on data file 7 for use by the Airmobile Model are loaded by the routine FIL7LD, described in this appendix.

2. ROUTINE FIL7LD:

a. Purpose. This routine loads the data contained on data file 7 that are used by the Airmobile Model.

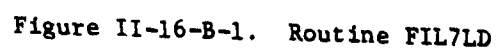
b. Input Variables. Standard common area, IFNT.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
EOH1(50)	DF7	Equipment item codes of all secondary priority items in the force; denoted by a one in the appropriate character location of this array. Four character codes are contained in each word.
JDATA(15)	DF7	Contents of the mission mix table for each escort/transport aircraft mix.
IFIL7(201-210)	DF7	Unit type designators of the forward rearm/refuel areas (FRRAs).
IFIL7(211-220)	DF7	Number of rearm points at the above FRRAs.
IFIL7(221-224)	DF7	Equipment item codes of the refuel devices.
IFIL7(231-234)	DF7	Number of nozzles at each refuel device specified in words 221-224.
IFIL7(241)	DF7	Maneuver time for refueling.
IFIL7(242)	DF7	Maneuver time for rearming.
ITABLE	DF7	Array where data is stored until it is put onto data file 7.
MKILL	DF7	Probability of kill data for air defense weapons.

d. Logical Flow (Figure II-16-B-1):

(1) Blocks 1 and 2. Bring in the file name table and print it.



II-16-B-2

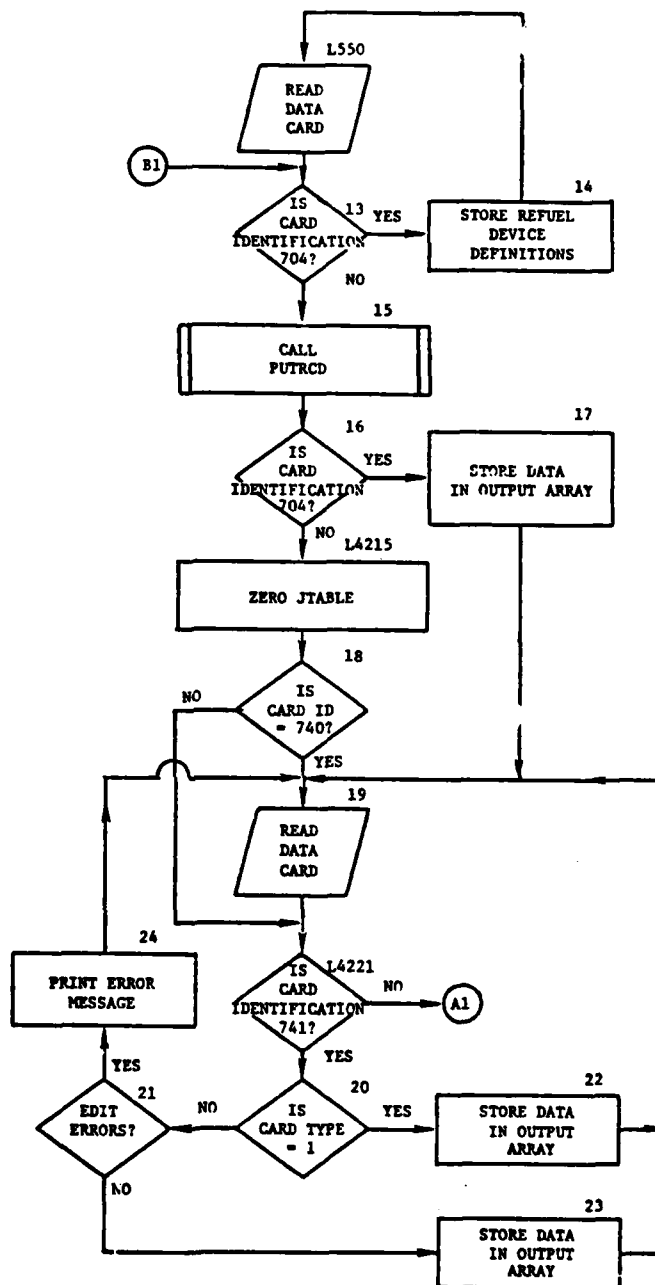


Figure II-16-B-1. Routine FIL7LD

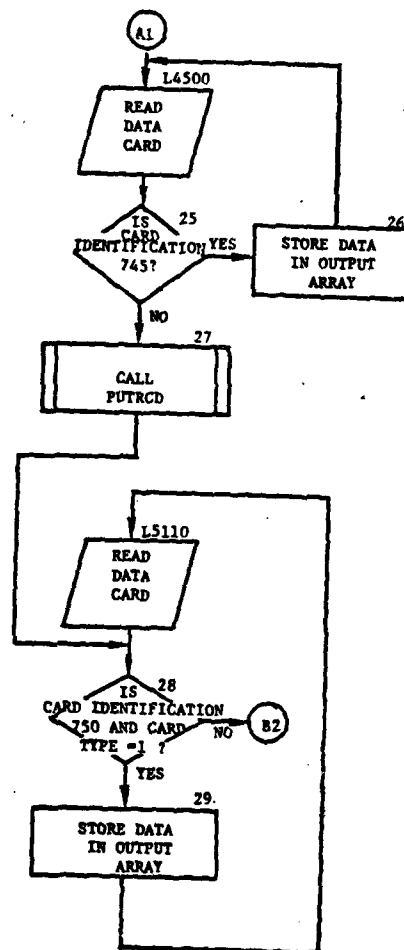


Figure II-16-B-1. Routine FIL7LD

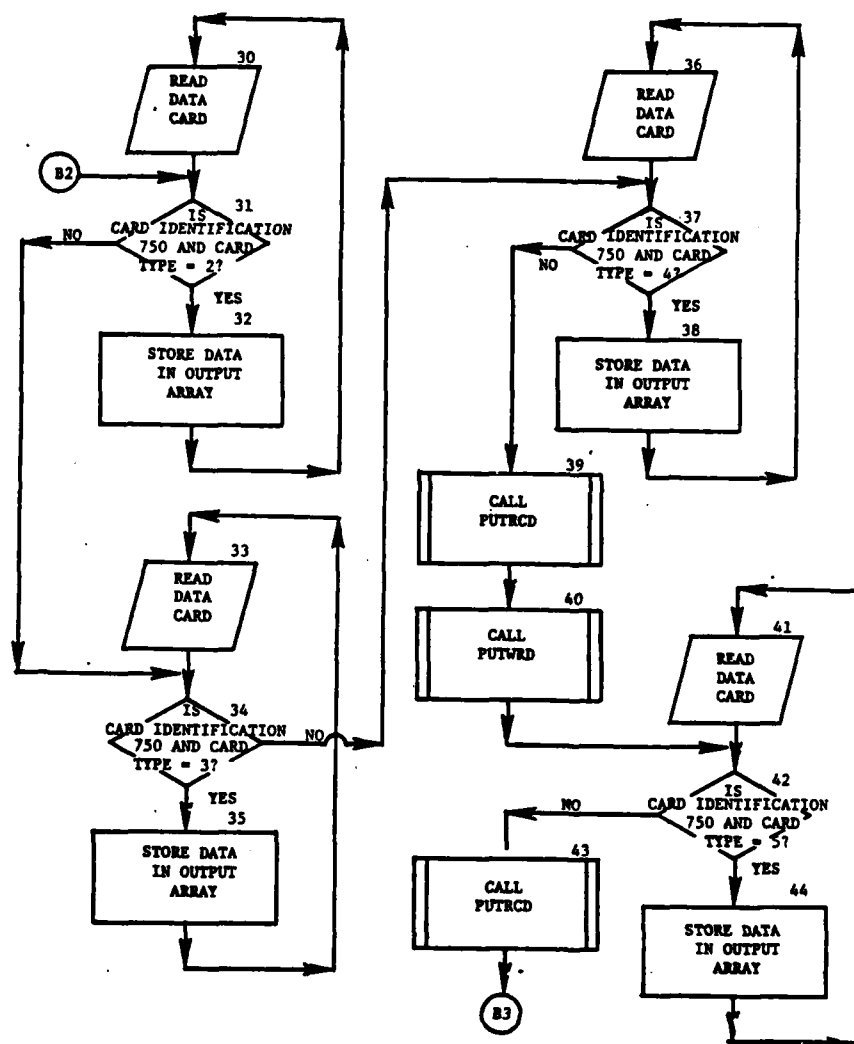


Figure II-16-B-1. Routine FIL7LD

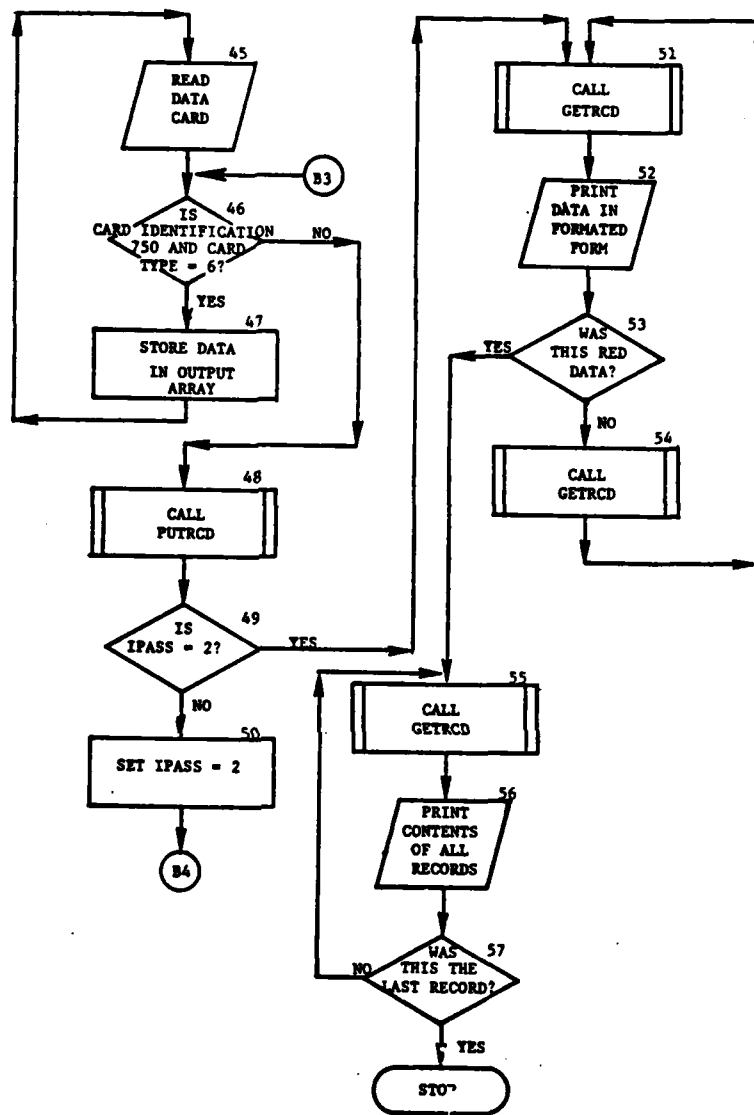


Figure II-16-B-1. Routine FIL7LD

II-16-B-6

(2) Blocks 3, 4, 5, and 6. Remove, create, and zero data files 7 and 24. Set IPASS equal to one to indicate that data for the Blue force are being processed.

(3) Blocks L200, 7, and 8. Read data cards. If the card identification is 701, set a flag in the output array for each secondary priority equipment type.

(4) Blocks L400, 9, and 10. The 702 cards are processed. They contain the mission mix data that are entered into the output array.

(5) Blocks L500, 11, and 12. After the 702 cards are processed, the 703 cards are read and the forward rearm and refuel area (FRRA) definitions that the cards contain are entered into the output array.

(6) Blocks L550, 13, 14, and 15. The 704 cards that contain the refuel device definitions are read and the data are stored in the output array. The data read from the four card types are put on the data file.

(7) Blocks 16, 17, L4215, 18, and 19. If the next card is a 740 card, the suppression mission data are read from it and stored in the output array. The JTABLE array is zeroed and, if the card was a 740 card, the next card is read in. If not control goes to block L4221.

(8) Blocks L4221, 20, 21, 22, 23, and 24. If the card read had data errors, a message is printed. If not, the acquisition data are stored in the appropriate place in the output array. When all 741 cards are processed, control passes to block L4500.

(9) Blocks L4500, 25, 26, and 27. If the card read is a 745 card, the aircraft data that it contains are stored in the output array. After all 745 cards are read the data from the 740, 741, and 745 cards are put on the data file.

(10) Blocks L5110, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, and 40. Card types 1 through 4 of card identification 750 are read. The air defense weapon data from each are stored in the output array. Each card is checked for identification and type. After all cards are read, the data are put on the data file.

(11) Blocks 41, 42, 43, and 44. The air defense weapon data for rotary wing aircraft are read from card type 5 of the identification 750 cards. The data are stored in the output array and, after all type 5 cards have been read, the data are put on the data file.

(12) Blocks 45, 46, 47, and 48. The air defense weapon data for fixed wing aircraft are read from card type 6 of the identification 750 cards. The data are stored in the output array and, after all type 6 cards have been read, the data are put on the data file.



(13) Blocks 49 and 50. If this was the second pass, control goes to the output portion of the program. If not, set IPASS equals two and read the data for the Red force, starting with the 701 cards.

(14) Blocks 51, 52, 53, 54, 55, 56, and 57. After the data have been loaded for both forces, print the contents of the first two records with descriptions of the contents. Then, print the contents of all data file 7 records.

## APPENDIX C

### OUTPUT DESCRIPTIONS FOR AIRMOBILE DATA INPUT

An 80-column image of each constant data input card is printed immediately after it is read. A sample of the listing produced in this manner is shown in Figure II-16-C-1. The listing of the data shown in Figure II-16-C-2 is generated by reading the first record and printing descriptive titles with the data. A similar listing for the Red force is generated from the second record. A word dump of all 52 records is also provided.

II-16-C-2

Figure 11-16-C-1. Sample of 80-Column Card Image, Airmobile Model

[illegible]

FRRA DEFINITIONS (UTD, NO. OF REARM POINTS)

VALH, 6	VELH, 6	, 0	, 0	, 0	, 0	, 0
---------	---------	-----	-----	-----	-----	-----

REFUEL DEVICE DEFINITIONS (EIC.NO. NOZZLES)

96.6 0.0 0.0 0.0

MANUEVER TIME FOR REFUELING	2
MANUEVER TIME FOR REARMING	2

**Figure II-16-C-2. Sample Data Listing, Airmobile Model**

NOTES

II-16-C-4

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR AIRMOBILE

(AVAILABLE UNDER SEPARATE COVER)

II-16-D-1

NOTES

II-16-D-2

## CHAPTER 17

### COMBAT SERVICE SUPPORT DATA INPUT PROGRAMS

1. INTRODUCTION. The Combat Service Support data input programs create and load the data file required by the Combat Service Support Model. The data loaded include tables defining consumption rates of various items of equipment, the weight and volume constraints and the load and unload times of the transportation vehicles, and the supply classes of the equipment items.

2. ROUTINES:

a. General. The Combat Service Support data input programs consist of the following routines.

b. Routine CSSLD. This routine controls the other routines of the Combat Service Support data input. It creates the file used to store the data, then calls the data load and file dump routines.

c. Routine LOAD11. This routine is responsible for the reading and editing of the Combat Service Support data cards. It also loads the data in the correct position on the file.

d. Routines EDITF1 and EDITF2. These routines are called by LOAD11 to edit fields one and two of the data card respectively. Field one contains the force indicator and field two contains an equipment item code.

e. Routine DUMP11. This routine provides both a dump of the Combat Service Support data file in tabular format and an unformatted record dump.

3. FILES. Data file 11 is used to contain the Combat Service Support data.



NOTES

II-17-2

## APPENDIX A

### COMBAT SERVICE SUPPORT DATA LOAD INPUT REQUIREMENTS

Complete descriptions of the constant data load input requirements for combat service support are documented in Appendix A, Combat Service Support Model Input Requirements, to Chapter 16 of the Period Processor (Section IV).

NOTES

II-17-A-2

## APPENDIX E

### COMBAT SERVICE SUPPORT DATA INPUT PROGRAM DESCRIPTIONS

1. INTRODUCTION. Loading of the Combat Service Support Model data is controlled by the routine CSSLD which, in turn, calls routine LOAD11 to read the input data cards and place data upon data file 11 and DUMP11 to take data from the file and provide a formatted data listing. LOAD11 is supported by a pair of card editing routines, EDITF1 and EDITF2.

#### 2. ROUTINE CSSLD:

a. Purpose. This routine controls the loading of Combat Service Support Model data on data file 11. Two routines are called; LOAD11 and DUMP11.

b. Input Variables. None.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
NERA	Print	Number of data errors detected.

#### d. Logical Flow (Figure II-17-B-1):

(1) Blocks 1, 2, and 3. After the file name table has been obtained, routine REMOVE is called to remove outdated data from data file 11. Routine CREATE is called to create data file 11 as 249 records of 512 words each. This is an expandable file and can be expanded by the SERSUP routine of the Combat Service Support Model.

(2) Block 4. After the file has been created, records 11 through 18 are filled with words containing 99999. Records 19 through 10 and 19 through 239 are filled with zeros.

(3) Block 5. Routine LOAD11 is called to load data file 11 from cards.

(4) Block 6. If no errors have occurred in LOAD11, a dump of data file 11 is made by routine DUMP11. The file name table is printed and control returns to the calling routine.

#### 3. ROUTINE LOAD11:

a. Purpose. This routine loads Combat Service Support Model data from cards onto data file 11.

b. Input Variables: Data card types 1101, 1102, 1103, and 1104.

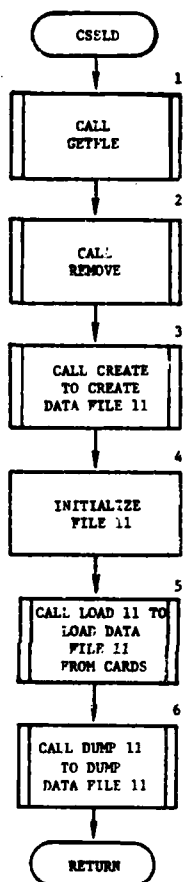


Figure II-17-B-1. Routine CSSLD

II-17-B-2

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
IREC19(425)	DF11	A value of zero, indicating no pending supply action entries, stored in record 29, word 425.
IREC19(426)	DF11	A value of one, indicating that this is the first 70-word block created on data file 11 for supply action entries, stored in record 29.
IREC19(427)	DF11	A value of 70, indicating this is first 70-word block created on data file 11 for supply action entries. When the second block is created, IREC19(425) and (426) are incremented by 70.
IREC19(428)	DF11	Blue force class I consumption rate, stored in record 29, word 428.
IREC19(429)	DF11	Red force class I consumption rate, stored in record 29, word 429.
ISUPB(11,201)	DF11	Weight, volume, and maximum of nine transport equipment item code preferences for each of 200 equipment items plus personnel for Blue force, stored in records 30 through 33 plus words 1 to 163 of record 34.
ISUPR(11,201)	DF11	Same as above--for Red force--stored in records 35 through 38 plus words 1 to 163 of record 39.
ICAPB(3,50)	DF11	Equipment item code, weight, volume, and capacities of up to 50 transport types for Blue force, stored in record 34, words 363 to 512.
ICAPR(3,50)	DF11	Same as above--for Red force--stored in words 363 to 512 of record 39.
IAXB(201,14)	DF11	Number of each item available for resupply for each day (14 days maximum) of battle for Blue force, stored in records 40 to 44 plus words 1 to 254 of record 45.
IAXR(201,14)	DF11	Same as above--for Red force--stored in records 46 through 50 plus words 1 to 254 of record 51.
ITIMEB(6)	DF11	Load and unload times for Blue force transports, stored in words 1 through 6 of record 29.
ITIMER(6)	DF11	Load and unload times for Red force transports, stored in words 207 to 212 of record 29.

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
ISCB(200)	DF11	Supply class for each Blue force equipment item, stored in words 7 to 206 of record 29.
ISCR(200)	DF11	Supply class for each Red force equipment item, stored in words 213 to 412 of record 29.

d. Logical Flow (Figure II-17-B-2):

(1) Block 1. Cards in the CSSLD data deck are read; the 80-column card images are printed and stored on a scratch file. Cards continue to be read until 9999 is found in columns 73 through 76. The scratch file is then rewound.

(2) Block 2. A card image is read from the scratch file. The form number, which is in columns 75 through 76, is read into IFORM. For Combat Service Support data an 11 should appear in columns 73 and 74. Columns 73 and 74 are stored in IFILE. If IFILE equals 11, this card contains Combat Service Support data and control passes to block 3. If IFILE is not equal to 11, it is checked to determine if it contains 99. If it does, this is the end of the data to be loaded, and control goes to block 9; otherwise, an error message is printed and control goes to block 3.

(3) Block 3. IFORM is checked to determine if it is valid; (i.e., greater than zero and less than or equal to four). If not, an error message is printed and control goes to block 2; otherwise, control goes to block 4.

(4) Block 4. If IFORM equals 4, control goes to block 8. If IFORM equals 3, control goes to block 7. If IFORM equals 2, control goes to block 6. If IFORM equals 1, control goes to block 5.

(5) Block 5. This is an 1101 card and contains transport load and unload times plus food consumption rates. Routine EDITF1 is called to ensure that the force designator code for this card is valid. If all transport times are zero or blank, a message is printed. If no consumption rates for class I are given, an error message is printed. If the force designator was invalid, an error message is printed. Blue class I consumption rate is stored in word 428 of record 429. Red class I consumption rate is stored in word 429 of record 429. The transport load and unload times are also stored in core in the ITIMEB or ITIMER arrays until all data are loaded. If no error occurs, it will be put on data file 11 in record 29. When IFORM changes, control returns to block 2.

(6) Block 6. This is an 1102 card and contains the equipment item codes (EIC) of the transports that will carry the consumable item for which the EIC is in columns 3 through 5. The weight and volume of the consumable item are also input in this card. The weight is in hundredths of pounds on the card, but is stored in tenths of pounds on data file 11. Routine EDITF1 is called to ensure that the card contains a valid force designator code. Routine EDITF2 is called to ensure that the consumable has a valid EIC. The supply class listed for the consumable is also edited to ensure that it is between one and ten. A check is made to ensure that if the consumable is

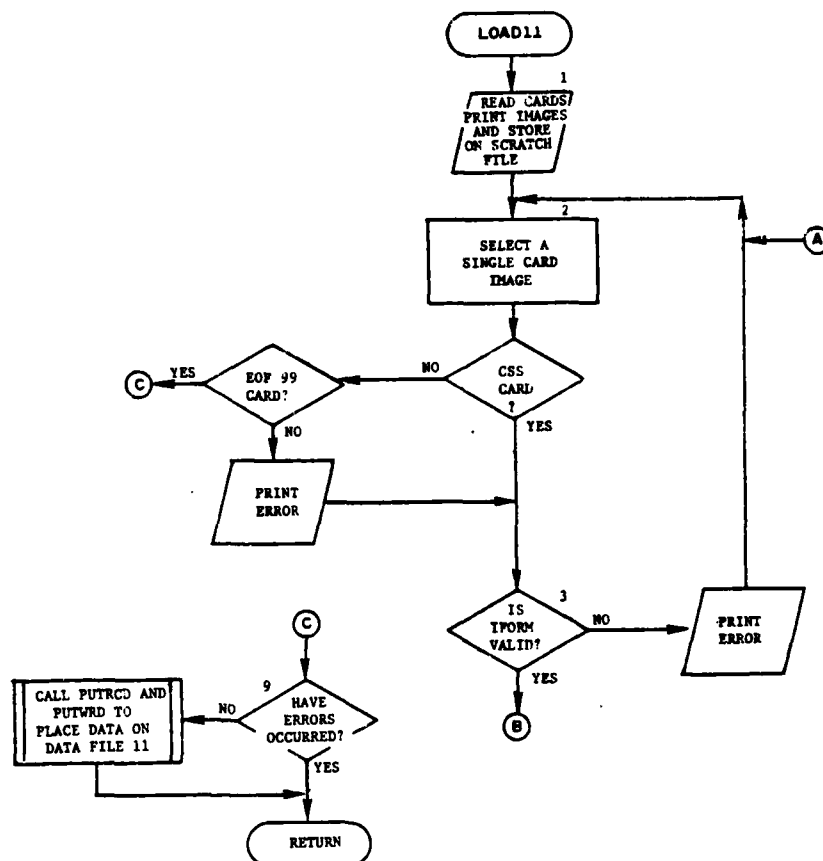


Figure II-17-B-2. Routine LOAD11 (Continued on Next Page)



AD-A112 658

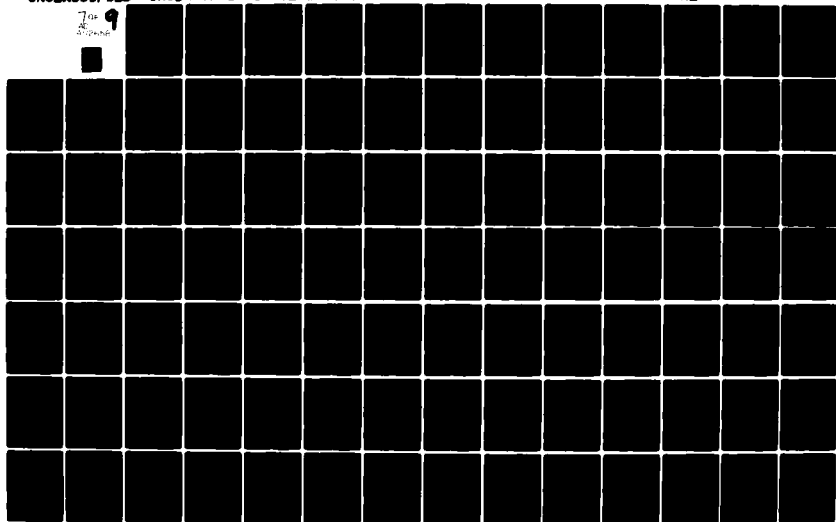
ARMY COMBINED ARMS COMBAT DEVELOPMENTS ACTIVITY FORT--ETC F/8 18/7  
DIWAS MODEL DOCUMENTATION, VOLUME II. PROGRAMMER/ANALYST MANUAL--ETC(U)  
JUL 76

UNCLASSIFIED

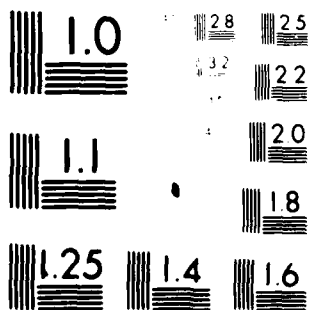
CACDA-TR-8-76-VOL-8-PT-1

ML

700  
9



11265



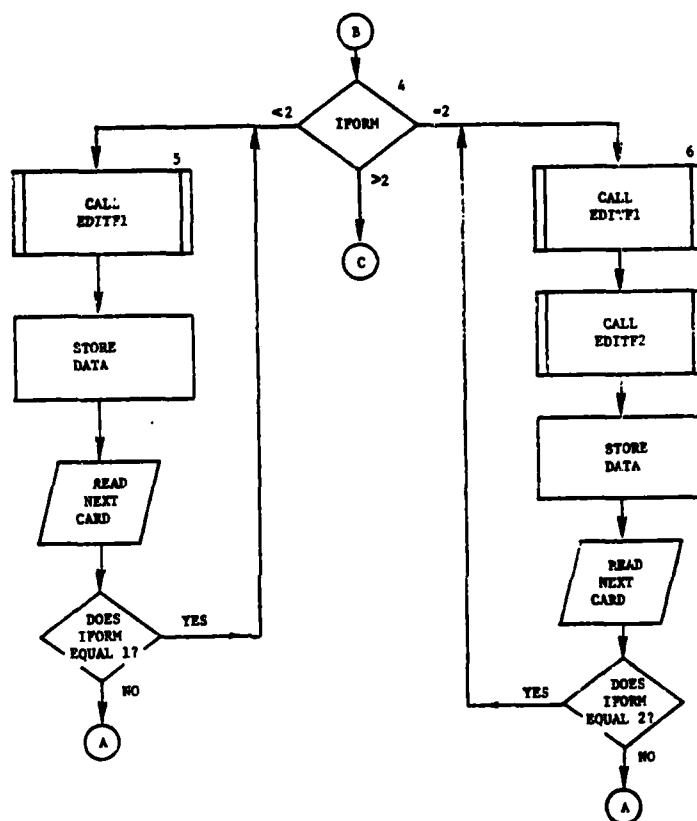


Figure II-17-B-2. Routine LOAD11 (Continued)

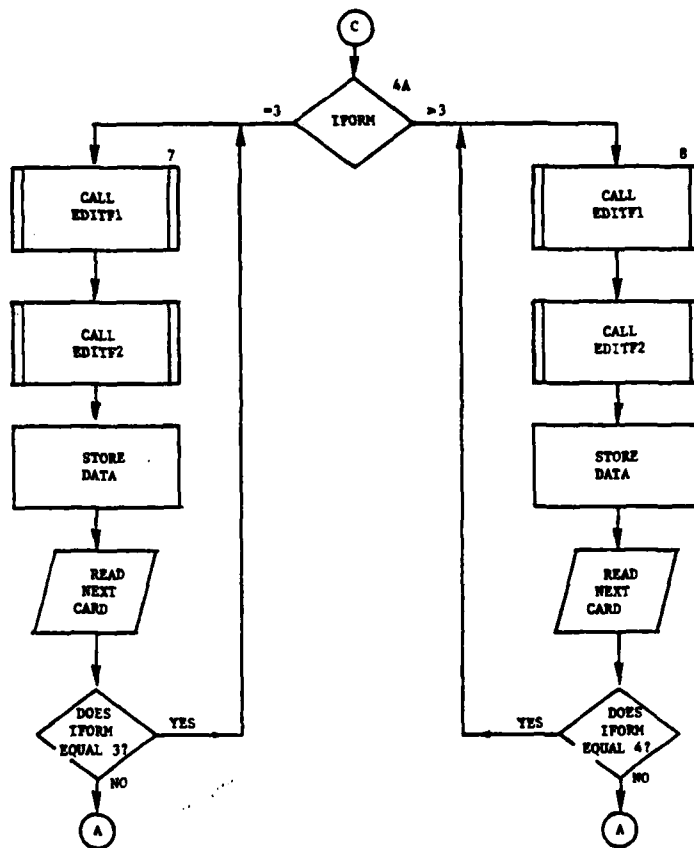


Figure II-17-B-2. Routine LOAD11 (Concluded)

class 7, a nonzero EIC is specified for the first transport choice under unit distribution. This EIC is used in determining how quickly personnel and major end items will travel to the receiving unit after they have been issued. All nine possible transport choices on this card are subjected to the EIC editing. Both the weight and volume of the consumable must be greater than zero. Messages are printed for all errors that are encountered. A count of the number of errors is also printed. All information is stored in core in the appropriate ISUPB or ISUPR arrays except the class of the consumable. The class is appropriately stored in ISCB or ISCR. When IFORM changes, control returns to block 2.

(7) Block 7. This is an 1103 card and contains the weight and volume capacities of the transports. A maximum of three transports can be entered on each card. Each transport is identified by its EIC. The weight capacities of the transports are read in hundredths of pounds but stored on data file 11 in tenths of pounds. Routine EDITF1 is called to check the validity of the force designator code. The EIC of each transport listed is checked for validity by routine EDITF2. If the volume capacity of a particular transport is input as 99999999.0 it is stored as the largest number that can be stored in a 24 bit integer word; (i.e., 8388607). The EIC, weight capacity, and volume capacity are stored in the appropriate ICAPB or ICAPR arrays. Messages are printed for all errors encountered. When IFORM changes, control returns to block 2.

(8) Block 8. This is an 1104 card and contains the number of major end items by type that are available to the division for resupply for each day of battle. The number of personnel available for replacement is also input on this card format. Routine EDITF1 is called to check the validity of the force designator code. Routine EDITF2 is called to verify the EIC for each card. The resources are stored in the appropriate IAVB or IAVR arrays. When IFORM changes, control returns to block 2.

(9) Block 9. At this point Combat Service Support data have been loaded in core. If no data load errors were encountered, the information is put on data file 11.

#### 4. ROUTINE EDITF1:

a. Purpose. This routine edits the force designator located in the second column of every input card.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
NERR	Common	Number of errors that have occurred in loading data.
IFD	Call	Force designator for card being read.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
NERR	Common	Number of errors that have occurred in loading data.
IFORCE	Call	Number indicating whether card pertains to Red or Blue Force.

d. Processing Description. This routine examines the force designator in column two of every input card. If it contains a B, IFORCE is set equal to one. If it contains an R, IFORCE is set equal to two. If it contains anything else, an error message is printed, the error counter is incremented, and control returns to routine LOAD11.

5. ROUTINE EDITF2:

a. Purpose. This routine edits the equipment item codes. It is called by routine LOAD11.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
NERR	Common	Number of errors encountered in loading data file 11.
IEIC	Call	Equipment item code to be edited.
LIM	Call	The largest valid integer value IEIC may have.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
NERR	Common	Number of errors encountered in loading data file 11.

d. Processing Description. This routine edits each equipment item code to ensure that it contains an integer value greater than zero but less than the value contained in LIM. If it contains a value outside of this range an error message is printed. NERR is incremented, and control returns to routine LOAD11.

6. ROUTINE DUMP11:

a. Purpose. This routine prints formatted tables displaying Combat Service Support Model data contained on data file 11 and then prints a dump of the file.

b. Input Variables: See output variables from routine LOAD11.

c. Output Variables: Printed data tables and dump of data file 11.

d. Logical Flow (Figure II-17-B-3):

(1) Block 1. In this block both Red and Blue force information is printed. This includes the class I consumption rates in pounds per man per day and the transport load and unload times for each distribution method.

(2) Block 2. Blocks 3 through 7 are performed first for the Blue force; then, for the Red force.

(3) Block 3. The supply class for each equipment item in the force is printed. If no supply class is given for a particular item, that item will not be resupplied. Supply classes are given numbers from 1 to 10.

(4) Block 4. The weight (pounds) and volume (cubic feet) of each consumable item to be resupplied is printed for this force. Each item is identified by its equipment item code.

(5) Block 5. The weight (pounds) and volume (cubic feet) capacities of each transport assigned to resupply consumables is printed for this force. Each transport is identified by its equipment item code. A maximum of 50 transports may appear.

(6) Block 6. The EICs of the transports assigned to deliver consumables for this force are printed. The first, second, and third preferences are displayed for each consumable under each of the three distribution methods.

(7) Block 7. This block prints the amounts of major end items and personnel that are available for resupply to the division during each day of battle.

(8) Block 8. If the Red force data were not printed, control returns to block 3 to process the Red force. Otherwise, a record-by-record dump of data file 11 is printed. Control then returns to routine CSSLD.

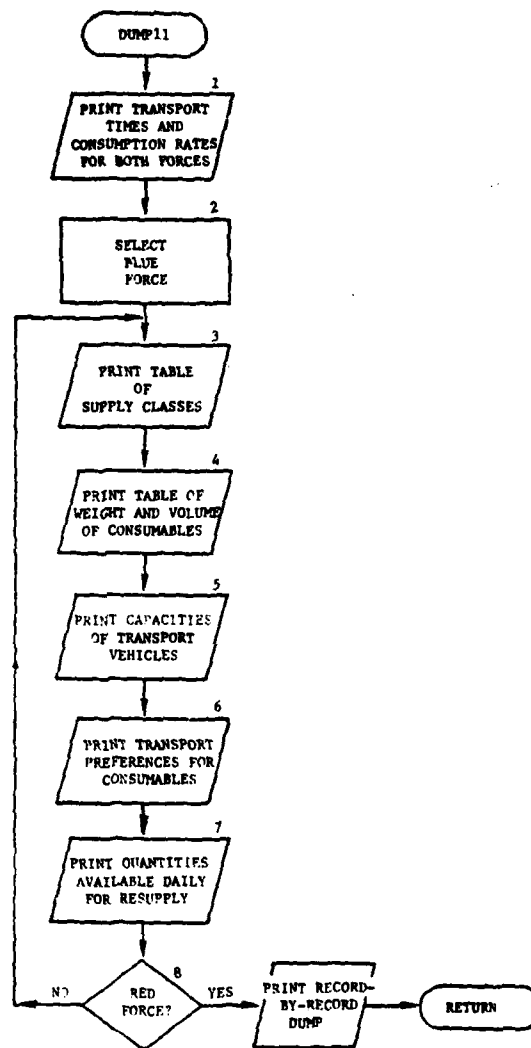


Figure II-17-B-3. Routine DUMP11

II-17-B-11



NOTES

II-17-B-12

## APPENDIX C

### COMBAT SERVICE SUPPORT DATA LOAD OUTPUT DESCRIPTIONS

1. INTRODUCTION. The constant data input printouts for Combat Service Support are listed in Figure II-17-C-1. The 80-column card image is the listing of all data entered in the files. The other printouts are used to check the values of data entered and to identify the originating card if necessary.

Printout Title	Data Source	Card ID	Card Type
80-Column Card Image	Card data deck	All	All
Elapsed Time to Handle Vehicles	Data file 11	1101	1
Consumption Rates	Data file 11	1101	1
Weights and Volumes of Consumables	Data file 11	1102	1
Weight and Volume Capacity of Transports	Data file 11	1103	1
Equipment Item vs Class of Supply	Data file 11	1102	1
Transport Preference for Hauling Consumables	Data file 11	1102	1
Consumables Available Daily for Resupply	Data file 11	1104	1
File Dump	Data file 11	All	All

Figure II-17-C-1. Combat Service Support Constant Data Printouts

2. EIGHTY-COLUMN CARD IMAGE. The 80-column card image printout is illustrated in Figure II-17-C-2. Across the top at the far left is the number 1, indicating card type 1. The letter B indicates Blue force data and the number 1101 is the card ID.

3. ELAPSED TIME TO HANDLE VEHICLES AND CLASS I CONSUMPTION RATE. The next printout in the constant data input printout series is illustrated by Figure II-17-C-3. At the upper half of the format is the elapsed time in minutes for handling the transports for both Blue and Red forces. The lower half of the figure has the subsistence consumption for both Red and Blue forces. The data for this printout are extracted from data file 11 and were originally enscribed on card ID 1101.

FILE-11 DATA CAPT IMAGES

18	60	1	30	50	7	5	1	1101	CAPT NO.	1
19	60	1	30	50	4	5	1	1101	CAPT NO.	2
18	1	1	100	03	161	164	0	161	164	3
18	2	1	620	20	165	166	162	162	166	4
18	3	1	642	21	165	166	162	162	166	5
18	4	1	100	04	164	167	161	167	161	6
18	5	1	100	02	164	167	161	167	161	7
18	6	1	100	02	164	167	161	167	161	8
18	7	1	04	01	164	167	161	167	161	9
18	8	1	40	01	164	167	161	167	161	10
18	9	1	100	02	164	167	161	167	161	11
18	10	1	1737	100	164	167	161	167	161	12
18	11	1	4000	109	164	167	161	167	161	13
18	12	1	1440	100	164	167	161	167	161	14
18	13	1	11500	309	164	167	161	167	161	15
18	14	1	6849	175	164	167	161	167	161	16
18	15	1	6849	175	164	167	161	167	161	17
18	16	1	6840	175	164	167	161	167	161	18
18	17	1	6000	150	164	167	161	167	161	19
18	18	1	12000	302	164	167	161	167	161	20
18	19	1	6500	200	164	167	161	167	161	21
18	20	1	10304	301	164	167	161	167	161	22
18	21	1	10304	301	164	167	161	167	161	23
18	22	1	21100	509	164	167	161	167	161	24
18	23	1	15800	350	164	167	161	167	161	25
18	24	1	50600	40009	162	164	0	162	164	26
18	25	1	10394	309	164	167	161	167	161	27
18	26	1	21100	509	164	167	161	167	161	28
18	27	1	60600	40009	162	164	0	162	164	29
18	28	1	10394	309	164	167	161	167	161	30
18	29	1	60600	40009	162	164	0	162	164	31
18	30	1	10394	309	164	167	161	167	161	32
18	31	1	60600	40009	162	164	0	162	164	33
18	32	1	10394	309	164	167	161	167	161	34
18	33	1	60600	40009	162	164	0	162	164	35
18	34	1	10394	309	164	167	161	167	161	36
18	35	1	60600	40009	162	164	0	162	164	37
18	36	1	10394	309	164	167	161	167	161	38
18	37	1	60600	40009	162	164	0	162	164	39
18	38	1	10394	309	164	167	161	167	161	40

Figure II-17-C-2. Combat Service Support 80-Column Card Image

ELAPSED TIME TO HANDLE TRANSPORTS (IN MINUTES)						
----- INVT DISTRIBUTION -----		----- SUPPLY POINT DISTRIBUTION -----		----- AIRLIFT -----		
LOAD TRANSPORTS	UNLOAD TRANSPORTS	STACT TRANSPORTS	LOAD TRANSPORTS	LOAD AIRCRAFT	UNLOAD AIRCRAFT	
BLUE 60	1	70	60	5	1	
RED 60	1	30	60	5	1	

CLASS I CONSUMPTION RATES	
BLUE FORCE	RED FORCE
7	4

Figure II-17-C-3. Elapsed Time to Handle Vehicles

4. WEIGHTS AND VOLUMES OF CONSUMABLES AND VEHICLES. Figure II-17-C-4 illustrates the format for this printout. One printout is for Red force and one for Blue. The titles of the columnar headings are identical with those of the card formats. At the bottom of that format is shown weight and volume capacity of vehicles. The data from the upper half of the illustration were taken from data file 11 and originated from card ID 1102. Data from the lower half of the illustration were also taken from data file 11 but were punched in card ID 1103.

5. CLASSES OF SUPPLY. Classes of supply printout format is shown in Figure II-17-C-5, Equipment Item vs Class of Supply. One printout is for Red force and the other for Blue force. The information for this printout was extracted from data file 11 and was punched in card ID 1102.

6. TRANSPORT PREFERENCE FOR HAULING CONSUMABLES. Figure II-17-C-6 depicts the format for this type of printout. One such table is printed for Blue and another for Red. This table lists the consumable at the far left and, for each consumable, preferred transports. Data for this printout were taken from data file 11 and were punched originally in card ID 1102.

7. MAJOR END ITEMS AND PERSONNEL AVAILABLE DAILY FOR RESUPPLY. This table, as illustrated in Figure II-17-C-7, provides the amounts of major end items and personnel available each day for resupply. These data were extracted from data file 11 and were input initially in card ID 1104.

# WEIGHT AND VOLUME OF BLUE CONSUMABLES

ITEM CODE	WEIGHT (LBS.)	VOLUME (GAL. FT.)
1	1.0	.03
2	6.2	.20
3	6.6	.20
4	1.0	.04
5	1.0	.02
6	1.0	.02
7	.5	.01
8	.4	.01
9	1.0	.02
22	100.0	1.00
24	700.0	25.00
25	20640.0	1154.70
26	300.0	25.00
27	17.7	1.00
28	640.0	20.40
29	10560.0	1044.00
30	40.0	1.00

# WEIGHT AND VOLUME CAPACITY OF BLUE TRANSPORT VEHICLES

ITEM CODE	WEIGHT (LBS.)	VOLUME (GAL. FT.)
97	27700.0	2000.00
101	16500.0	700.00
102	7140.0	240.00
107	24500.0	2704.00
108	24000.0	2714.00
109	31000.0	1000.00
110	11100.0	700.00
117	16550.0	700.00
90	4000.0	1000.00

Figure II-17-C-4. Weight and Volume of Consumables and Capacity of Vehicles

# CLASSIFIED BY BLUE EQUIPMENT ITEMS

ITEM CODE	SUPPLY CLASS	ITEM CODE	SUPPLY CLASS	ITEM CODE	SUPPLY CLASS
41	4	44	7	81	7
42	7	45	5	82	7
43	7	46	5	83	7
44	4	47	0	84	7
45	5	48	0	85	7
46	5	49	0	86	7
47	5	50	0	87	7
48	5	51	0	88	7
49	5	52	0	89	7
50	5	53	0	90	7
51	5	54	5	91	7
52	5	55	5	92	7
53	5	56	5	93	7
54	5	57	5	94	7
55	5	58	5	95	7
56	5	59	5	96	7
57	5	60	5	97	7
58	5	61	5	98	7
59	5	62	5	99	7
60	5	63	5	100	7
61	5	64	5	101	7
62	5	65	5	102	7
63	5	66	5	103	7
64	5	67	5	104	7
65	5	68	5	105	7
66	5	69	5	106	7
67	5	70	5	107	7
68	5	71	5	108	7
69	5	72	5	109	7
70	5	73	5	110	7
71	5	74	5	111	7
72	5	75	5	112	7
73	5	76	5	113	7
74	5	77	5	114	7
75	5	78	5	115	7
76	5	79	5	116	7
77	5	80	5	117	7
78	5	81	5	118	7
79	5	82	5	119	7
80	5	83	5	120	7

Figure II-17-C-5. Equipment Item versus Class of Supply

# TRANSPORT PREFERENCES FOR BLUE CONSUMABLES

CONSUMABLE			UNIT DISTRIBUTION			SUPPLY POINT DISTRIBUTION			APPLY		
ITEM	CODE	ITEM	CODE	ITEM	CODE	ITEM	CODE	ITEM	CODE	ITEM	CODE
1	161	161	0	161	164	161	164	161	164	161	164
2	165	165	162	162	165	162	165	162	165	162	165
3	165	165	162	162	165	162	165	162	165	162	165
4	164	164	161	161	164	161	164	161	164	161	164
5	164	164	161	161	164	161	164	161	164	161	164
6	164	164	161	161	164	161	164	161	164	161	164
7	164	164	161	161	164	161	164	161	164	161	164
8	164	164	161	161	164	161	164	161	164	161	164
9	164	164	161	161	164	161	164	161	164	161	164
10	164	164	161	161	164	161	164	161	164	161	164
11	164	164	161	161	164	161	164	161	164	161	164
12	164	164	161	161	164	161	164	161	164	161	164
13	164	164	161	161	164	161	164	161	164	161	164
14	164	164	161	161	164	161	164	161	164	161	164
15	164	164	161	161	164	161	164	161	164	161	164
16	164	164	161	161	164	161	164	161	164	161	164
17	164	164	161	161	164	161	164	161	164	161	164
18	164	164	161	161	164	161	164	161	164	161	164
19	164	164	161	161	164	161	164	161	164	161	164
20	164	164	161	161	164	161	164	161	164	161	164
21	164	164	161	161	164	161	164	161	164	161	164
22	164	164	161	161	164	161	164	161	164	161	164
23	164	164	161	161	164	161	164	161	164	161	164
24	164	164	161	161	164	161	164	161	164	161	164
25	164	164	161	161	164	161	164	161	164	161	164
26	164	164	161	161	164	161	164	161	164	161	164
27	164	164	161	161	164	161	164	161	164	161	164
28	164	164	161	161	164	161	164	161	164	161	164
29	164	164	161	161	164	161	164	161	164	161	164
30	164	164	161	161	164	161	164	161	164	161	164
31	164	164	161	161	164	161	164	161	164	161	164
32	164	164	161	161	164	161	164	161	164	161	164
33	164	164	161	161	164	161	164	161	164	161	164
34	164	164	161	161	164	161	164	161	164	161	164
35	164	164	161	161	164	161	164	161	164	161	164
36	164	164	161	161	164	161	164	161	164	161	164
37	164	164	161	161	164	161	164	161	164	161	164
38	164	164	161	161	164	161	164	161	164	161	164
39	164	164	161	161	164	161	164	161	164	161	164
40	164	164	161	161	164	161	164	161	164	161	164

Figure II-17-C-6. Transport Preference for Hauling Consumables



# BLUE CONSUMABLES AVAILABLE DAILY FOR RESUPPLY

CONSUMABLE	ITEM	CONF	DAILY AMOUNTS														TOTAL
			DAY 1	DAY 2	DAY 3	DAY 4	DAY 5	DAY 6	DAY 7	DAY 8	DAY 9	DAY 10	DAY 11	DAY 12	DAY 13	DAY 14	
21	0	1	0	0	1	1	0	1	1	0	0	0	0	0	0	0	3
24	0	1	0	1	1	1	0	1	1	0	0	0	0	0	0	0	4
27	2	2	1	1	1	1	1	1	1	0	0	0	0	0	0	0	9
87	0	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	3
AA	2	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	9
186	2	2	2	1	1	1	1	1	1	0	0	0	0	0	0	0	10
PECS	20	220	220	220	120	120	120	120	120	0	0	0	0	0	0	0	1040

Figure II-17-C-7. Blue Major End Items and Personnel Available Daily for Resupply

APPENDIX D

SOURCE LISTINGS FOR CONSTANT DATA INPUT PROCESSOR COMBAT SERVICE SUPPORT

(AVAILABLE UNDER SEPARATE COVER)

NOTES

II-17-D-2

SECTION III

ORDERS INPUT PROCESSOR

III-2

## CHAPTER 1

### GENERAL DESCRIPTION OF THE ORDERS INPUT PROCESSOR

1. INTRODUCTION. In the conduct of a war game with the DIVWAG system, the Orders Input Processor is a primary point of interface between the war gaming staff and the Period Processor. It provides the means by which the gaming staff instructs simulated units to carry out desired activities. Although explicit unit portrayal and assessment of unit actions, together with limited decision making processes, are automated within the Period Processor, the gaming staff maintains its ultimate control of the course of the game through the Orders Input Processor.

2. INPUT PROCESSOR SUBSYSTEMS. The Orders Input Processor is composed of two independently run computer subsystems: the DIVWAG Scenario Language (DSL) Compiler and the Operating Instructions (OPERINS) Loader.

a. DSL Compiler. The DSL Compiler converts instructions written in a specialized source language into tables which, in turn, are used to drive the Period Processor. DSL orders are combined into unit scenarios, where each unit scenario contains the series of orders to be followed by one specified unit, or into battle paragraphs, which are used to control the Ground Combat Model of the Period Processor. DSL orders must be provided at the beginning of each period of simulated combat. The length of a period of simulated combat is under control of the gaming staff and is set by the gaming staff as part of the DSL input.

b. OPERINS Loader. The OPERINS Loader allows the gaming staff to interject, at the beginning of a period of simulated combat, certain data items for the control of ground-based sensors, for the control of fire support allocation, and for the introduction of intelligence from sources beyond the scope of sources internal to the Period Processor. Operating Instructions must be provided for the first period of simulated combat within a game. Their use thereafter is left to the discretion of the gaming staff.

NOTES

III-1-2

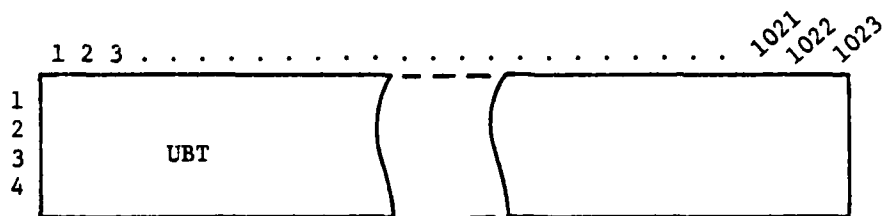
## CHAPTER 2

### THE DIVWAG SCENARIO LANGUAGE COMPILER

1. GENERAL DESCRIPTION. The DIVWAG Scenario Language (DSL) compiler is designed to recognize statements composing a unit scenario or battle paragraph and translate them into tables which are easily used by the Period Processor. One table is generated for each unit scenario or battle paragraph, and is stored on the DSL data file. A unit battle table (UBT), which serves as a directory to the unit scenario and battle paragraph tables, is generated and is also stored on the file. The table is described in detail in Figure III-2-1. Each of the unit scenario and battle paragraph tables contains 21 columns and as many rows as necessary. The orders, modifiers, and conditionals are stored in the tables by use of numerical codes and implicit location within the table. The numerical codes of the orders (NORD) are listed in Figure III-2-2, and those associated with the conditionals (NCON) are shown in Figure III-2-3. Figures III-2-4, through III-2-7 list the meanings associated with each column as a function of NORD or NCON. In the unit scenario tables, rows beginning with NORDs and NCONs may appear in almost any order. In the battle paragraph tables the first five rows are reserved for the list of units participating in the battle. These are stored ten UIDs per row (using two columns per UID) leaving column 21 unused. The remainder of the table will have rows beginning with an NCON intermingled with rows of labels of unit scenario statements. These labels are associated with the unit scenario order the unit is to execute if the battle is terminated by the preceding conditional. They are stored one per column, again leaving column 21 unused.

2. DESCRIPTION OF PROCESSING. The limited vocabulary of orders, modifiers, and conditional clause elements and the rigid syntax of the language allow the decomposition of a statement to be accomplished using a small number of logical decisions. This process is depicted in Figure III-2-8. The first level of logic determines if the statement is a COMMENT, FINIS, unit scenario identification (ID), battle paragraph identification (BATTLE), unit order, or battle description statement. If the statement is one of the first four types listed, processing may be completed without further decision making. If the statement is a unit order, the second level of logic determines if it contains a conditional. If a conditional is present, a third level of logic is invoked to determine the conditional clause type. A different set of third-level logic is used to determine the type of order. If the order may have modifiers associated with it, a fourth level of logic determines which modifiers are present. If the statement is part of a battle description, another set of second level logic determines if it contains the list of units participating in the battle, or a conditional followed by a list of labels designating the next orders that the units are to execute if the battle is terminated by this conditional. If it contains the list of units, no other logic decisions are necessary. If not, the conditional will be processed by the third-level logic mentioned above. The final processing of any statement may consist of merely transferring execution to the appropriate place in the code (e.g., comment or FINIS statement), but usually requires the extraction and storage of data (e.g., UID, coordinates, munition type, labels).





- . The unit battle table is a 4 x 1023 word array.
- . Each of the 1023 columns may contain information about a unit scenario or battle paragraph.
- . Unit scenario information is stored in column 1 to UNTPT.
- . Battle paragraph information is stored in columns BATPT to 1023. with information from the first battle paragraph encountered being stored in column 1023, information from the second stored in column 1022, etc.
- . Unit scenario information description:
  - Row 1 contains the first four characters of the unit identification
  - Row 2 contains the last four characters of the unit identification
  - Row 3 contains the number of records (NR) required to contain the unit scenario and the pointer (PKNT) to the record containing the order currently being executed by the unit (value =  $NR + 1000 * PKNT$ )
  - Row 4 contains the record number of the first record used to store the unit scenario.
- . Battle paragraph information description:
  - Row 1 contains the first four characters of the battle identification
  - Row 2 contains the last four characters of the battle identification
  - Row 3 contains the number of records (NR) required to store the battle paragraph and the number of units (NU) participating in the battle (value =  $NU + 1000 * NR$ )
  - Row 4 contains the record number of the first record used to store the battle paragraph.

Figure III-2-1. Unit Battle Table Description

DSL Order	NORD
Accept Transport	20
Advance	6
Airmobile Assault	41
Assignment is	23
Assume control of	24
Breach	43
Build	42
Detach	26
Engage	15
Fire	9
Fire on targets of opportunity	10
Fly	7
Go to	-1
Join	25
Loiter	32
Mission is	39
Move	3
Prepare	2
Reconnoiter	8
Release transport	31
Remove	44
Retain	40
Stay	1
Stop task	45
Terminate	37
Withdraw	5

Figure III-2-2. Numerical Order Codes

Conditional Type	NCON
Assessed	-2
At location	-4
Class 3	-5
Class 5	-6
Cloud cover	-14
Equipment type	-7
Firing	-8
Fog index	-21
Halted at	-11
Moving	-9
Precipitation index	-16
Present strength	-12
Relative humidity	-18
Stopped	-10
Temperature	-15
Temperature gradient	-17
Time	-3
Visibility index	-13
Wind direction	-20
Wind speed	-19

Figure III-2-3. Conditional Type Codes

Column	Description
1	NORD
2	Time associated with the order
3	Air speed
4	Altitude
5,6	X and Y coordinates of designated ground zero
7	Number of rounds or number of volleys
8	Munition type
9	Impact radius
10	Height of burst
11,12	Unit identification of cooperating unit
13,14	Battle identification
15,16	X and Y coordinates of movement objective
17	Travel mode mnemonic, or reconnoiter by code
18	Movement priority
19	Unit width
20	Unit depth
21	Relative position in table of next order to be executed
See Figure III-2-5 for airmobile orders and Figure III-2-6 for engineer orders.	

Figure III-2-4. Standard Order Column Descriptions

Column	Description
1	NORD (= 20, 39, 40, or 41)
3	Game time associated with order
9	Number of aircraft $\leq 0$ Number of trips $\leq 0$
10	Mix table code, or aircraft item code
12	Number of escorts, or target number
15,16	X and Y coordinates of the objective
21	Relative position in table of next order to be executed
	All other columns are unused.

Figure III-2-5. Airmobile Order Column Descriptions

Column	Description
1	NORD (= 42, 43, 44, or 45)
3	Barrier = 0 Bridge or facility = 1
4	First 3 characters of barrier code
5	Last 3 characters of barrier code
6	Priority code
7	Complete task by time = 0 Begin task by time = 1 (See column 9)
8	Completion of task is desired = 0 Completion is mandatory = 1
9	Game time associated with the task
21	Relative position in table of next order to be executed
	All other columns are unused.

Figure III-2-6. Engineer Order Column Descriptions

Column	Description
1	NCON (less than -1)
2,3	UID associated with the conditional
4	Time associated with the conditional
5,6	X and Y coordinates of the location associated with the conditional
7	Quantity data
8	Percentage data
9	True = negator appeared in conditional False = no negator
10	Less than = -1: equal = 0: greater than = +1
11	Item code of equipment in conditional
20	Relative location in table of next order to be executed if conditional is false
21	Relative location in table of next order to be executed if conditional is true
	All other columns are unused.

Figure III-2-7. Conditional Column Descriptions

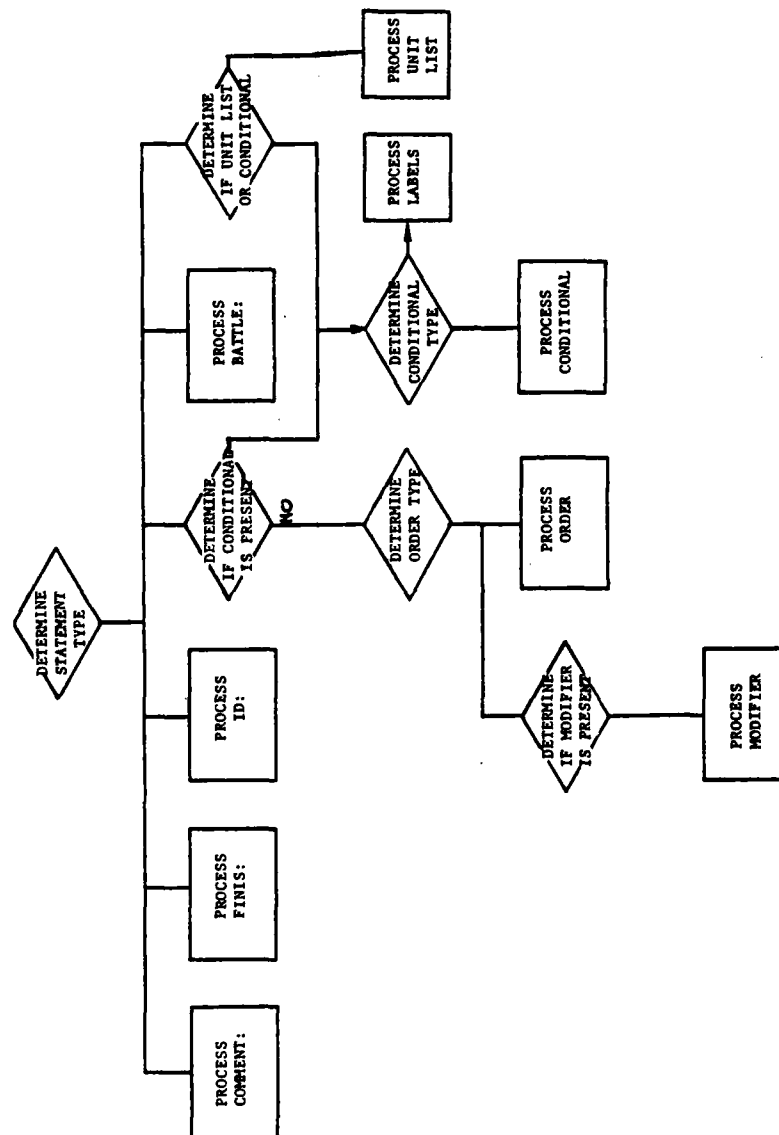


Figure III-2-8. DSL Compiler Logical Flow



3. DSL LOGICAL FLOW. The order in which the routines are invoked to process each DSL statement is depicted in Figure III-2-9. As indicated by this figure, DSLCMP serves as the controlling routine of the compiler. It first invokes DSLINT which initializes the dynamic arrays and variables and processes the DSL control statements. RDSTMT is used to read all DSL data cards. It is called by DSLINT for the processing of the first three DSL data cards and by DSLCMP for the processing of the remainder of the data deck. DSLCMP performs a preliminary examination of each statement read to determine if it is a

a FINIS statement, a battle paragraph or unit scenario identification statement, or a battle paragraph or unit scenario descriptive statement. BATORD is called to further process battle paragraph descriptive statements, and UNTORD is invoked if the statement describes a unit scenario. Both BATORD and UNTORD call COND if a conditional is discovered in the statement. The decoding and storage of UIDs of units participating in a battle and the labels of the orders they are to execute immediately following a battle is accomplished by BATORD. After calling COND, if necessary, UNTORD determines the type of order contained in the statement and calls the appropriate routine. Each of the routines--AIRMOB, ENGR, FIRE, FLY, MOVE, STAY and TRANS--contains a list of the modifiers that may be used with the orders for which they are responsible. As each order is received, an array containing the modifiers that may be associated with it is set up and passed to KRAKMD. KRAKMD further decodes the statement by searching out the modifiers present and storing the associated data.

4. ROUTINE DESCRIPTIONS. A description of the routines comprising the DSL compiler follows. The purpose of each routine and the manner in which it accomplishes that purpose is included in the description.

a. Routine DSLCMP. This is the driving routine of the DSL compiler. Its first task is to call routine DSLINT which performs the initialization of the dynamic variables and processes the DSL control cards. Once that is accomplished, a loop involving the reading and decoding of the remaining statements is entered. RDSTMT reads each data card and performs initial processing. The statement is tested to determine if it is a comment. If it is, no further processing is necessary and the next statement is read. If the statement is the FINIS statement, the end of the DSL orders has been reached. PASS2 is called to complete processing of the tables. If debug print was requested, DUMPF is called to print the tables generated; otherwise, the statement is checked to determine if it is part of a unit scenario. If this is the case, UNTORD is called to process it. If the statement is part of a battle paragraph BATORD is invoked to decode it. If the statement is a battle paragraph or unit scenario identification statement, STOW is called to write the table associated with the battle paragraph or unit scenario just processed to the data file. If the statement cannot be recognized an error message is written and the next statement is processed.

b. Routines BLOCKD and DSLINT. These routines accomplish the initialization of the arrays and constants required by the DSL compiler. BLOCKD initializes the arrays of order names and codes, modifier names, array limits, and integer and special character constants. DSLINT sets the initial values of variables, such as the line counter, the error counters, and the file name table array. It calls RDSTMT to read the DSL control cards and stores the period time parameters input on those cards. It also initializes the unit

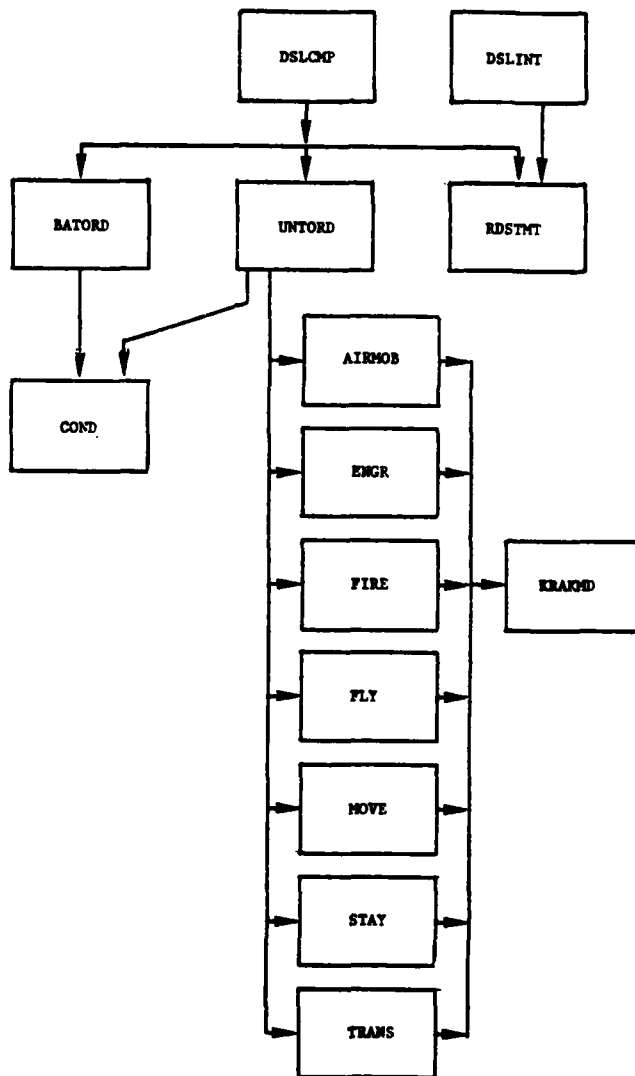


Figure III-2-9. DSL Logical Flow

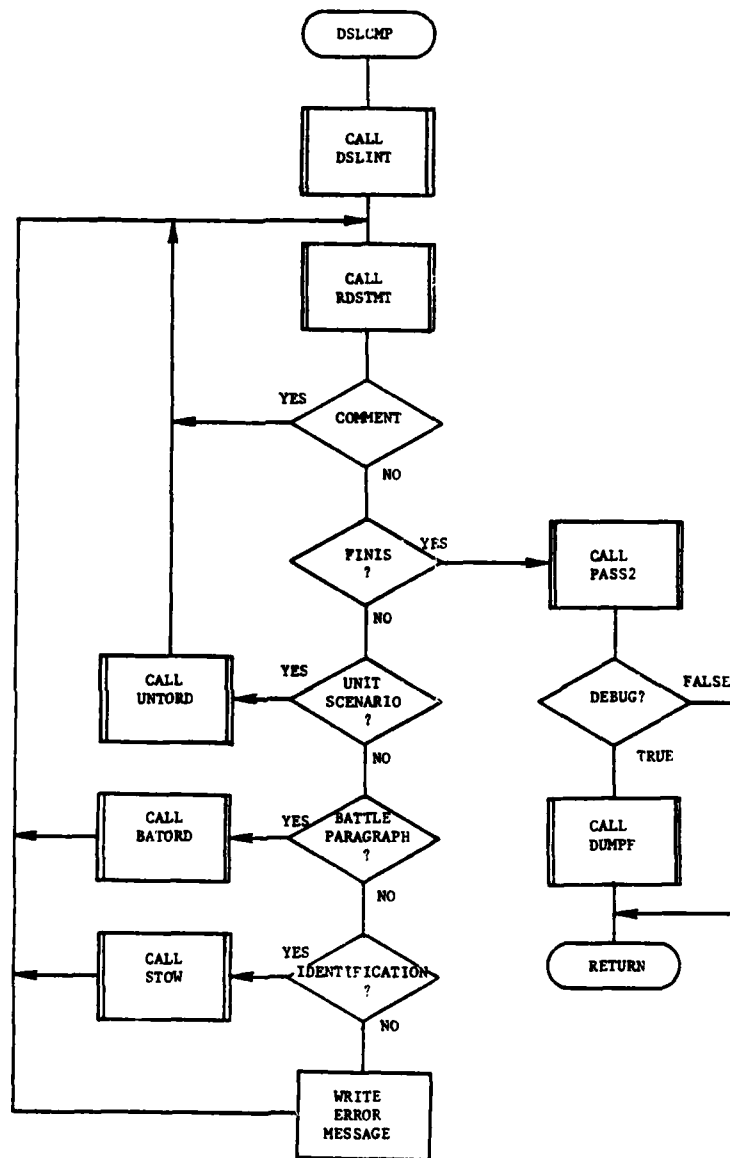


Figure III-2-10. Routine DSLCMP Logical Flow

location and unit identification tables used by the compiler, and fills the unit-battle directory table with blanks and zeros.

c. Routine RDSTMT. RDSTMT is responsible for the reading and initial processing of all data cards input to the compiler. Blanks are ignored in the statements, so the nonblank characters are extracted from each card and stored in a compressed format as each card is read. Each nonblank character is also checked to determine if it is a period, colon, or comma. The locations of those characters, if encountered, are stored as they are useful in decomposing the statement. The contents of each card are also printed. To keep the listings of the Blue force and Red force unit scenarios and the battle paragraphs separate, each ID and BATTLE statement is checked and the printer skips to the top of the next page if the statement does not belong to the same group of the last listed statement.

d. Routine BATORD. This routine processes statements belonging to a battle paragraph. These statements must be one of two types. The first type contains the list of UIDs of units participating in the battle. The UIDs are extracted from the statement and stored in the order array, and the number of UIDs is stored in the unit battle table (UBT) in the area assigned to this battle paragraph. The second type of statement is a conditional followed by a list of labels. These labels correspond to labels in the unit scenarios of the units in the battle. When a battle is terminated by a conditional, the units execute the orders designated by the list of labels following the conditional. Routine COND is called to process the conditional portion of the statement, and BATORD extracts the labels from the statement and stores them in the order array.

e. Routine UNTORD. UNTORD is called to process unit scenario descriptive statements. These statements are orders to the unit and can be preceded by a label, a conditional, or both. If the statement contains a colon, the characters preceding the colon are stored as the label. If the statement contains a conditional, routine COND is called to process it. The remainder of the statement is an order that is chosen from the DSL vocabulary of orders and that order's modifiers. The general type of the order is determined (e.g., moving, firing) and the appropriate order processing routine is called (See Figure III-1).

f. Routine COND. The routine COND is responsible for the decomposition of and storage of data from the conditional portion of the statements. There are 14 conditional vocabulary elements including 10 clauses and 4 types of data. The syntax allows seven clause types to be formed from these elements. Clause types can be connected by AND or OR with the limitation that both AND and OR cannot be used in the same statement. Routine COND determines if AND or OR is present, and processes each clause type independently. A clause type is processed by first determining if the unit identification element is present. If it is, and the next element is a unit activity (e.g., MOVING, AT LOCATION), the processing is completed; otherwise, the clause type cannot be recognized. If the unit identification element is not present, a search is made for a unit possession element. If this element is found processing is completed. If a unit possession element is not present, a search is made for a weather conditional element. If this element is present it is processed; otherwise, the clause type cannot be recognized. If the unit identification element is not

present, a time element is searched for; and if found, the time is stored; otherwise, a weather element is searched for indicating a weather conditional at a location if it is present and an unrecognizable clause type if it is not.

g. Routines AIRMOB, ENGR, FIRE, FLY, MOVE, STAY, and TRANS. These routines build an array of modifiers for the orders for which they are responsible and assure that the modifiers required by the orders are present in the statement. This is accomplished by building an array of modifiers that may be used with a particular order, passing this array to KRAKMD, and checking it when it is returned. If a required modifier or an exclusive modifier is missing, an error message is written.

h. Routine KRAKMD. This routine decomposes the modifier portion of the DSL statement and stores data associated with each modifier. Modifiers exclusively contain alpha characters. Most modifiers are followed by numerical data. Each modifier is processed independently by scanning the statement from the last character processed to the next numerical character or the end of the statement. If no modifier is found, an error message is written. If a modifier is present, the data associated with it (if any) is extracted and stored, and a search for another modifier begins. As each modifier is found, its position in the array of modifiers is set to zero, thereby disallowing more than one occurrence of the same modifier. Each coordinate pair, of orders having coordinate pairs (move, fly, advance, withdraw, reconnoiter, and airmobile assault), generates a separate order in the order table. After processing one of these order modifiers, the data stored in the first order generated are copied into the other orders. In the case of a reconnoiter or airmobile assault order, the last order generated is indicated.

i. Routine STOW. This routine is called by DSLCMP to write a unit scenario or battle paragraph to the data file. If a unit scenario is being stored, the labels associated with the scenario are written on the file in the five records immediately preceding the first record used to store the unit scenario. STOW also stores the number of statements (records on the file) in the appropriate location in the UBT and increments the order file record pointer.

j. Routine PASS2. PASS2 is the second pass of the DSL compiler. The labels used to identify orders in a unit scenario that are referenced later by the pseudo order GO TO or a battle paragraph statement are stored in character form. An array of labels is built as the unit scenario is processed. The position in the array corresponds to the order's position in the unit scenario. When a label is referenced by a GO TO or battle paragraph statement, it is stored in the appropriate position in the order array. PASS2 scans each unit scenario and battle paragraph for referenced order labels, matches labels found with a position in the unit scenario's array of labels, and substitutes the numerical position of the label for the characters in the order table.

k. Routine DUMPF. This routine is called by DSLCMP if the debug option on the DSL control cards is exercised. It provides a formatted dump of the tables built by the compiler by reading the contents of the order file and printing them under the appropriate format. Included in the dump is the period start time and period length, start of game/start of period flag, UBT pointers, resume of the formats used in printing unit scenarios, and an index of the unit scenarios listed. This is followed by a listing of the unit scenarios and battle paragraphs.

l. Routine TIME. The time input with a DSL order may be in either of two forms: a six-digit integer with the pairs of digits indicating days, hours, and minutes; or one to three integers followed by an appropriate day, hour, or minute indicator. Routine TIME converts the quantity input into an integer value, which is the number of centiminutes from the beginning of the period to the time indicated.

m. Routine CHCKID. This routine performs requested checks on unit and battle identifications. It can be requested to determine if a unit exists or is a resolution unit, whether the unit identification contains eight characters with the first one being B or R, or if the unit identification is a duplicate. A battle identification can be checked for duplication, for length (eight characters or less), or for existence in the list of battles.

n. Routine MVCHAR. This routine uses system mask and shift functions available to FORTRAN users to reposition characters.

o. Routines PAGE, IUIDF, MATCH, FINDN, EXPAND, IBCD, and RBCD. PAGE determines if the value of the line counter is greater than or equal to the number of lines allowed per page. If so, a header is written at the top of a new page and the line counter is reset. IUIDF is a function routine used to determine the position of a given UID in the Period Processor's list of UIDs. MATCH is used to determine if a given character string is a subset of a second character string (e.g., determines if a statement contains a particular order or modifier). If it is found to be a subset, the location is returned. FINDN will locate the next numerical character in a character string. EXPAND is used to move the four characters stored in one word to the first character in each of four words. Routines IBCD and RBCD are used to convert a string of numerical characters to its equivalent integer or floating point value.

NOTES

III-2-16

## APPENDIX A

### INPUT REQUIREMENTS FOR DIVWAG SCENARIO LANGUAGE COMPILER

1. INTRODUCTION. The essential components of the DIVWAG Scenario Language (DSL), as of any language, are a vocabulary; or set of symbols, words, or phrases with specified meanings; and a syntax, or set of rules for combining vocabulary elements. With DSL the vocabulary is limited to a small set of elements, and the syntax is relatively rigid; that is, few syntactic options exist. Within the DSL syntax, the highest level of composition is the Unit Scenario and the Battle Paragraph. This level parallels the paragraph of the English language, generally defined as being composed of one or more sentences and dealing with one point. A Unit Scenario contains the DSL commands for one unit within the DIVWAG system, and a Battle Paragraph contains the definition and instructions for one battle within the Ground Combat Model of the Period Processor. As the paragraph of the English language generally contains an introductory sentence followed by a number of elaborating sentences, so must a Unit Scenario or a Battle Paragraph contain an introductory section and an elaborating body. As a sentence of the English language generally expresses one thought and concludes with appropriate end punctuation, so must each piece of a Unit Scenario or a Battle Paragraph. As punctuation is critical to a proper understanding of written English, it is also critical to proper use of DSL. DSL differs from a language used for human communication in one highly significant aspect, semantic flexibility. In the English language a sentence could be interpreted as having several meanings, and a single thought could be expressed in a multitude of ways. In DSL, an order has only one meaning, and there is only one way to express an order.

#### 2. DSL COMPOSITION:

a. General. The highest level of composition within the DSL syntax is the Unit Scenario or the Battle Paragraph. Each contains an introductory section and an elaborating body. Each statement of the introductory sections or of the elaborating body is similar to a sentence of the English language and must end with a period. Any other use of a period within DSL is illegal; thus, decimal points are not permitted and, although abbreviations are possible within the vocabulary, they are not denoted by the period. The building blocks for a statement are order clauses, conditional clauses, and labels. In the following paragraphs, an order clause is represented by the symbol O, a conditional clause by the symbol C, and a label by the symbol L. Where examples are given below, the prototype order clause STAY UNTIL 011200 will be used, the prototype conditional clause TIME LESS THAN 011830 will be used, and the prototype label ABC will be used. Statements may not exceed 400 characters in length.

b. Unit Scenarios. A Unit Scenario contains the set of DSL orders to be followed by one unit through the course of the game period for which the DSL is prepared. A Unit Scenario contains one introductory statement and at least one elaborating statement.



(1) Unit Identification. The unit identification is the first statement of each Unit Scenario. It contains the characters ID: followed by the eight-character unit identification (UID) of the unit to follow the orders contained in the scenario. For example:

ID:B1234ABC.

The colon and period must be present as in the example. The UID must be composed of exactly eight alphanumeric (letters of the alphabet or Arabic numerals) characters, the first of which must be B or R.

(2) Commands. Commands comprise the elaborating body of a Unit Scenario. Each unit identification statement must be followed by at least one command. There is no practical limit on the number of commands which may appear within one Unit Scenario. A command is composed of an optional label, one or more optional conditional clauses, and exactly one mandatory order clause.

(a) Command Syntax. A command may be written in one of the four following basic forms:

O.  
L:O.  
IF C, THEN O.  
L:IF C, THEN O.

Using the prototypes introduced above, these forms are in actual DSL examples:

STAY UNTIL 011200.  
ABC:STAY UNTIL 011200.  
IF TIME LESS THAN 011830, THEN STAY UNTIL 011200.  
ABC:IF TIME LESS THAN 011830, THEN STAY UNTIL 011200.

The following rules of syntax must be observed:

1. If a label is present, it must be the first element of the command.
2. If a label is present, it must be followed by a colon.
3. If a conditional clause is present, it must be stated before the mandatory order clause and after the label (if used).
4. If a conditional clause is present, it must be introduced by the keyword IF and it must be followed by a comma and the keyword THEN, in that order.
5. The order clause must be the final clause of a command.

6. The order clause must be followed by a period.

(b) Syntax for Conditional Expansion. The basic command forms containing conditionals may be expanded to contain more than one conditional in the following forms:

IF C AND IF C AND IF C, THEN O.  
L:IF C AND IF C AND IF C, THEN O.  
IF C OR IF C OR IF C, THEN O.  
L:IF C OR IF C OR IF C, THEN O.

In using the conditional expansion feature, the following additional rules of syntax must be followed:

1. The logical keywords AND and OR may not both appear in one command.

2. The first conditional clause is introduced by the keyword IF; all following conditional clauses are introduced by the keywords AND IF or OR IF.

3. The last conditional clause is followed by a comma and the keyword THEN.

(c) Command Components:

1. The order clause within a command must be composed of one of the orders within the DSL vocabulary with any appropriate modifiers.

2. The conditional clauses within a command must be composed of conditionals contained within the DSL vocabulary with appropriate modifiers.

3. The label within a command must be composed of not more than three alphanumeric characters. Each label within one Unit Scenario must be unique. Labels are used to identify specific commands within a Unit Scenario. The use of labels is treated at length in paragraph 6. The keyword ID may not be used as a label since it is associated with unit identification.

(3) Unit Scenario Execution. A Unit Scenario contains all commands to be followed by the unit identified in the unit identification statement during the game period for which the scenario was written. A unit will follow commands only if it is a resolution unit and if it has a personnel strength of at least one mar. (It is assumed that the reader is familiar with the concept of resolution units within the DIVWAG system.) Since a nonresolution unit is not actually active within the Period Processor, any Unit Scenario for a nonresolution unit will be ignored. The Unit Scenario for any resolution unit having less than one man is also ignored.

(a) Sequencing. The first command in a Unit Scenario is always acted upon first. When execution of one command is completed, execution of the succeeding command is initiated. This direct sequencing of commands can be overridden by the proper use of labels and the GO TO order, or the Battle Paragraph, as discussed in paragraph 6.

(b) Execution of One Command:

1. Conditionals. Upon receipt of a command which contains a conditional clause or clauses, the condition is checked prior to execution of the order clause. If the condition stated in the conditional clause does exist, the order clause is executed. If the condition stated in the conditional clause does not exist, the order clause is ignored and the next command is executed. Processing of a command wherein the expanded conditional feature is used is identical except that all conditional clauses are checked. If the logical AND keyword is used, the condition stated in each conditional clause must exist for execution of the order clause. If the logical OR keyword is used, the order clause is executed if the condition stated in any one (or more) of the conditional clauses exists. In either case, the next command is executed if the appropriate conditional criteria are not met.

2. Order Clause. In the absence of conditional clauses or the presence of appropriate conditional criteria, the order clause of a command is executed. Duration of the event initiated by an order clause depends on the nature of the clause. Generally, the Unit Scenario is not reentered until execution of the event initiated by the order clause has been completed. Exceptions are discussed in paragraph 6.

(4) Unit Scenario Preparation. Unit Scenarios are written on standard program coding sheets in free form; that is, the first character can be written in any column of the form, and spacing between characters or a group of characters can be used to improve readability. As many lines as needed may be used for one command, each line representing a punched card. Each command must begin with a new line (new card). The information on the form is then key punched on standard cards, and the cards are assembled in proper sequence to become unit scenarios within the source deck.

c. Battle Paragraphs. One Battle Paragraph serves to identify each battle grouping (battle) within the Ground Combat Model of the Period Processor, to identify the units involved in that battle, to establish the conditions for termination of the battle, and to identify the command within their respective Unit Scenarios that each involved unit is to follow upon termination of the battle. A Battle Paragraph contains two introductory statements and at least one elaborating statement.

(1) Battle Identification (BID). The battle identification is the first statement of each Battle Paragraph. It contains the characters BATTLE: followed by the name associated with the battle (BID). For example:

BATTLE:BULGE.

The colon and period must be present as in the example. The battle identification (BID) must be composed of not more than eight alphanumeric characters; and, within one game period, each BID must be unique.

(2) Battle Declaration. The battle declaration is the second statement of each Battle Paragraph. It contains the phrase SURFACE UNITS:, followed by a list of the UIDs of units which are to be actively engaged in the battle or are to respond to battle conditionals contained in the Battle Paragraph. For example:

SURFACE UNITS:B123ABCD,B111222B,RAB12345.

The colon must be present, the UIDs must be separated by commas, and the period must be present, as in the example. No more than 34 UIDs may be used, and no more than 17 UIDs on either side (Blue or Red) are permitted.

(3) Battle Conditionals. Battle conditionals comprise the elaborating body of a Battle Paragraph. Each battle declaration statement must be followed by at least one battle conditional. There is no practical limit on the number of battle conditionals in one Battle Paragraph. A battle conditional is composed of one conditional clause and a series of labels, where the number of labels must equal the number of units listed in the battle declaration statement.

(a) Battle Conditional Syntax. A battle conditional must be written in one of the following forms:

WHEN C, THEN L,L,L.

WHEN C, AND WHEN C, AND WHEN C, THEN L,L,L.

WHEN C, OR WHEN C, OR WHEN C, THEN L,L,L.

or, using the prototypes listed above form one becomes:

WHEN TIME LESS THAN 011830, THEN ABC,ABC,ABC.

The following rules of syntax must be observed:

1. The logical keywords AND or OR may not both appear in one command.
2. The first conditional clause is introduced by the keyword WHEN; all following conditional clauses are introduced by the keywords AND WHEN or OR WHEN.
3. The last conditional clause is followed by a comma and the keyword THEN.
4. The number of labels must agree with the number of units listed in the battle declaration statement.

5. Labels must be separated by commas.
6. The final label must be followed by a period.
7. Each unit in the battle declaration statement must have a Unit Scenario.
8. The nth listed unit in the battle declaration must have in its Unit Scenario a command which has the nth label listed in the battle conditional.

(b) Battle Conditional Execution:

1. Timing. The list of battle conditional statements within a Battle Paragraph is reviewed after each simulation increment of the named battle within the Period Processor.

2. Sequence. Battle conditionals within a Battle Paragraph are executed in the sequence in which they appear in the paragraph.

3. Condition Met. If, in the course of executing battle conditionals, the condition stated in a given conditional clause does exist, the battle is terminated (another battle increment is not scheduled in the Period Processor); and each unit listed in the battle declaration statement is immediately scheduled to execute the command within its own Unit Scenario which has a label matching the label of that battle conditional associated with that unit. Association of units in the battle declaration and labels in the battle conditional is by position; that is, the nth listed unit is associated with the nth listed label.

4. Condition Not Met. If at the time of execution, the condition listed in a battle conditional is not met, processing continues with the next battle conditional. When all conditionals in a Battle Paragraph have been processed, and none of the stated conditions exists, a new battle increment is scheduled in the DIVWAG Period Processor.

3. DSL ORDER CLAUSES:

a. General. The previous paragraph introduced the order clause and the conditional clause as components of a DSL statement. This paragraph presents the elements of a DSL order clause, and paragraph 4 presents elements of a DSL conditional clause.

(1) Each order clause is composed of exactly one basic order, a variable number of order modifiers, and a variable number of data elements. In the following discussion, elements of DSL will always be written in upper case letters to differentiate from the accompanying narrative.

(2) The DSL Compiler does not recognize English words. It recognizes DSL elements. For example, the compiler recognizes the basic order FIRE as one DSL element. It also recognizes the basic order FIRE ON TARGETS OF OPPORTUNITY as one DSL element. The user must take care to provide the total DSL element when preparing DSL orders.

(3) Order modifiers may be mandatory, exclusive, or optional. A mandatory order modifier is one which must appear each time the basic order with which it is associated appears. A set of exclusive order modifiers is a group of modifiers, exactly one of which must appear each time the associated basic order appears. An optional order modifier is one which may appear with the associated basic order but is not required by the DSL Compiler.

(4) A data element may be associated with a basic order or with an order modifier. Data elements are never optional; that is, where a data element is associated with a basic order or an order modifier, the data element must be present.

(5) The first element of an order clause is always a basic order. If a data element is associated with a basic order, it is always the second element of an order clause. Order modifiers and associated data may be written after the basic order and its data element (if any) in any sequence, with the limitation that a data element associated with any modifier must immediately follow the modifier. Punctuation, or any entry not specifically identified as part of a DSL element, is not allowed.

b. DSL Basic Orders. Basic DSL orders are grouped within six generic categories: stay, move, engage, engineer, transfer, and pseudo orders. These categories contain basic orders for which the simulated activity is generally similar. In most cases, the various orders generate different secondary actions which are discussed with the appropriate basic order.

(1) Stay Activity Orders. Units with a stay activity are motionless. They will consume class I and class III or IIIA supplies. If on the ground, they may be assessed by area fires and by aerial attack and, if air defense units, may fire upon enemy aircraft. Other capabilities of units with stay activity orders depend upon the specific order.

(a) STAY. The STAY order requires one of the exclusive modifiers FOR (a specified period of time) or UNTIL (a specified time). For example:

STAY FOR 1 HOUR.  
STAY UNTIL 010230.

If the unit is an artillery unit it will enter the TACFIRE Model of its division automatically and will fire upon targets if so directed by TACFIRE. This is also the default order. If a resolution unit has no Unit Scenario, or if all commands in its Unit Scenario are completed, the Period Processor will automatically generate a STAY UNTIL (end of the game period) for the unit.

(b) FIRE ON TARGETS OF OPPORTUNITY. This order should only be given to artillery units. It requires the same modifiers as the STAY order and, for an artillery unit, has the same effect as a STAY order. Examples:

FIRE ON TARGETS OF OPPORTUNITY FOR 30 MIN.  
FIRE ON TARGETS OF OPPORTUNITY UNTIL 011230.

(c) PREPARE. The PREPARE order requires one of the exclusive modifiers FOR (a specified period of time) or UNTIL (a specified time) and allows the optional modifiers AT WIDTH (unit front in meters) -DEPTH (unit depth in meters). For example:

PREPARE UNTIL 031500.  
PREPARE FOR 12 HOURS AT WIDTH 1500 - DEPTH 500.  
PREPARE AT WIDTH 2000 - DEPTH 775 FOR 1 DAY.

The PREPARE order causes the unit to assume a defensive posture at its current location. If unit dimensions are specified, these are used. If unit dimensions are not specified, those loaded in the data base for a defensive posture are used. This is one of the three orders under which a unit may become involved in battle within the Ground Combat Model of the Period Processor. Rules for such engagement are discussed under the ENGAGE order. An artillery unit given the PREPARE order does not enter the TACFIRE Model.

(d) LOITER. The LOITER order requires the three modifiers: FOR (a specified period of time), AT ALTITUDE (a specified altitude in feet), and AT SPEED (a specified flight speed in knots). For example:

LOITER FOR 15 MIN AT ALTITUDE 3000 FT AT SPEED 30 KNOTS.

The LOITER order should be given to aircraft units only. The unit will remain at its current location and specified altitude for the specified period of time. Flight speed is used to determine fuel consumption. Since an aircraft unit under the LOITER order is not vulnerable to air defense, the order should only be used for units over friendly territory.

(2) Move Activity Orders. Move activity orders direct surface and air movement. The general requirement of these orders is specification of the route the unit is to follow. The unit is moved from its current location to each point specified within the order clause along a straight line. Class I and class III or IIIA are generally consumed, and ground units are generally vulnerable to area fire and aerial fire. Vulnerability of air units depends on the order. Other actions depend upon the specific order.

(a) MOVE. The MOVE order is the prototype order for ground movement. The modifier TO (series of X-Y coordinates) is mandatory, and the modifiers BY (travel mode mnemonic), PRIORITY (movement priority = 1,2,3, or 4), and AT WIDTH-DEPTH (unit width and depth in meters) are optional. For example:

MOVE TO 0123000-0096500.  
MOVE BY TCCM TO 0115000-0095750,  
0115000-0098500, 0117500-0100000  
Priority 3.  
MOVE TO 0116000-0172500 BY TCCM AT WIDTH 3000-DEPTH 2000.  
MOVE PRIORITY 1 to 0098000-0118800.

The MOVE order causes ground movement over the specified route. Movement mode used by the DIVWAG Movement Model is as specified or, if the BY modifier is not present, cross country deployed movement is assumed. The priority code, used by the DIVWAG Engineer Model in allocation of engineer resources should an obstacle be encountered on the move, is set to 4 if the PRIORITY modifier is not present. If not specified in the DSL order, the unit width and depth dimensions will be assigned according to the units activity as loaded in the pregame data base.

(b) WITHDRAW. The WITHDRAW order uses the same modifier structure as the MOVE order with one exception. For WITHDRAW and ADVANCE orders the modifier BY must specify a 4 digit integer movement rate (kilometers/hour) in place of 4 character movement mnemonic. For example:

WITHDRAW TO 0117000-0098000 AT WIDTH 2000-DEPTH 880.  
WITHDRAW AT WIDTH 3000-DEPTH 500 BY TCCD  
TO 0088000-0115000, 0098000-0117000,  
0102500-0118000 PRIORITY 1.  
WITHDRAW TO 0124000-0100500 AT WIDTH 2000-DEPTH 1000 BY 0033.

The WITHDRAW order combines effects of the MOVE and PREPARE orders. It causes a unit to assume a withdrawal posture while accomplishing ground movements. This is one of the three orders under which a unit may become engaged in battle within the DIVWAG Ground Combat Model. Engagement procedures are discussed under the ENGAGE order.

(c) ADVANCE. Modifier requirements of the ADVANCE order are identical to those of the WITHDRAW order with the restriction that only one pair of coordinates may be specified. For example:

ADVANCE TO 0101500-0101888.  
ADVANCE BY TCCR AT WIDTH 1200-DEPTH 1000 TO 0123400-0090500.  
ADVANCE TO 0122000-0110000 BY 0066.

The ADVANCE order is the third order under which a unit may become engaged in ground combat. The unit assumes an attacking posture and proceeds toward the specified objective which, for proper ground combat engagement, should be to the rear of an opposing maneuver unit. Engagement procedures are discussed under the ENGAGE order.



(d) FLY. In using the FLY order, the three modifiers OVER (series of X-Y coordinates) AT SPEED (air speed in knots) AT ALTITUDE (altitude in feet) are mandatory. For example:

FLY AT SPEED 125 AT ALTITUDE 2500  
OVER 0111000-0088000.

This order is issued to an air unit to cause it to move along a flight path from its present location to a specified coordinate point. A list of coordinates following the modifier OVER is used to describe end points of all legs of the flight path. The air unit will be located (land) at the last point listed. FLY orders should be reserved for administrative movement over nonhostile territory.

(e) AIRMOBILE ASSAULT. In writing this order the modifier TO (series of X-Y coordinates) is mandatory, and the modifier AT TIME (specified time) is optional. For example:

AIRMOBILE ASSAULT TO 0111000-0111000,0123000-  
0111000.  
AIRMOBILE ASSAULT AT TIME 030500 TO  
009530-0088000.

This order causes the airmobile assault segment of the DIVWAG Airmobile Model to be activated. The airmobile task force conducts an airmobile movement along the specified flight path. Initiation of movement is scheduled such that leading elements of the airmobile task force arrive at the final coordinate if this is possible considering flight speeds, distance, and time the order is received. If the AT TIME modifier is not used, movement is initiated when the order is received. The airmobile task force is vulnerable to air defense weapons. To properly execute the order, the airmobile task force must have previously received an ACCEPT TRANSPORT order. Techniques for inclusion of this order in a Unit Scenario are discussed in paragraph 6.

(f) RECONNOITER. In writing the RECONNOITER order, the four modifiers BY (reconnaissance control code), OVER (series of no more than seven X-Y coordinates), AT SPEED (flight speed in knots) and AT ALTITUDE (altitude in feet) are all mandatory. For example:

RECONNOITER BY H322 OVER 0123000-0122000,  
0123000-0117000, 0133000-0117000,  
0133000-0122000 AT SPEED 50 AT  
ALTITUDE 100.

Processing of the order varies depending upon the nature of the unit receiving the order and the reconnaissance control code.

1. If the first character of the reconnaissance control code is the letter A, the unit receiving the order must be an Air Force reconnaissance control flight. The entire unit conducts the mission flying at the specified altitude and speed over the specified route. The fourth character of the reconnaissance control code specifies the sensor load. Sensors are activated upon receipt of the order and remain active over the entire flight path.

2. If the first character of the reconnaissance control code is the letter M, the unit receiving the order must be an army surveillance aircraft (Mohawk type) flight. The entire unit conducts the flight as outlined in the previous paragraph. The fourth character of the reconnaissance control code identifies the sensor package (which must include the side looking airborne radar (SLAR) with a ground terminal), and the second and third characters control range, delay, and direction of the SLAR.

3. If the first character of the reconnaissance control code is the letter H or F, the unit receiving the order must be an army unit containing light observation helicopters (H) or light fixed wing scout aircraft (F). In response to the order, a mission unit is created to conduct the mission. The mission unit flies to the first coordinate, at which point target sensing may commence. Upon completion of the mission, the mission unit returns to the unit which received the order. The mission unit may conduct a route or an area reconnaissance, as determined by the control code and discussed in paragraph 6.

4. In all cases, the RECONNOITER order provides the ability to control airborne sensors. Aircraft responding to a RECONNOITER order are vulnerable to air defense weapons.

(3) Engagement Orders. The group of engagement orders permit the game staff to control application of firepower within the Period Processor. The FIRE order controls both the Area Fire and Nuclear Assessment Models; the ENGAGE order controls the Ground Combat Model; the MISSION IS order controls the Air Ground Engagement Model. It is axiomatic that these orders be given only to appropriate units. FIRE orders should be given only to units capable of delivering conventional or nuclear indirect fires. ENGAGE orders should be given only to maneuver units capable of engagement within the Ground Combat Model. MISSION IS orders should be given only to units capable of carrying out a close air support mission either with attack helicopters or tactical Air Force aircraft.

(a) FIRE. The FIRE order activates the Area Fire or Nuclear Assessment Model. It should, of course, appear within the Unit Scenario of a unit that is capable of carrying out the fire mission. The FIRE order structure depends upon whether a conventional or nuclear fire is required.

1. For conventional fires the FIRE order must contain the mandatory modifiers ON (coordinates of target) and MUNITION TYPE (Area Fire munition code beginning with A) and one of the exclusive modifiers, NUMBER OF ROUNDS (a number), NUMBER OF VOLLEYS (a number), or IMPACT RADIUS (a number). For example:

FIRE ON 0123000-0098500 NUMBER OF ROUNDS  
25 MUNITION TYPE A013 IMPACT  
RADIUS 100.

The unit receiving the order will fire the specified number of rounds or volleys of the type specified upon the specified location.

2. For nuclear fires, four modifiers should be viewed as mandatory. These are ON (desired ground zero), MUNITION TYPE (munition code starting with N or D), NUMBER OF ROUNDS 1, and HEIGHT OF BURST (preset height of burst option index). IMPACT RADIUS (desired height of burst if not preset) is required if HEIGHT OF BURST is not preset. The desired ground zero is specified in terms of one X-Y coordinate pair. The munition code is a four-character alphanumeric code wherein the first character must be N or D and the remaining characters relate to the Nuclear Assessment Model constant data base, allowing specification of a firing weapon, a munition, a fuzing option, and a yield. Specification of a height of burst depends upon whether the specified fuze allows free selection of height of burst or only preset height options. If preset options are used, the modifier HEIGHT OF BURST is followed by a height index (1, 2, 3, 4). If free selection of a height of burst is possible, the HEIGHT OF BURST is followed by the number zero; and IMPACT RADIUS, by desired height in meters. Examples:

FIRE ON 0127000-0115000 NUMBER OF ROUNDS 1  
MUNITION TYPE NA33 HEIGHT OF BURST 2  
IMPACT RADIUS 100.

FIRE ON 0127000-0115000 NUMBER OF ROUNDS 1  
MUNITION TYPE NXA2 HEIGHT OF BURST 1  
IMPACT RADIUS 500.

(b) ENGAGE. The ENGAGE order has one mandatory modifier, IN BATTLE (battle name), and is always used in conjunction with the ADVANCE order. For example:

ADVANCE TO 0123100-0125000.  
ENGAGE IN BATTLE BRAVO.

The combination of an ADVANCE order followed by an ENGAGE order is used to initiate a battle within the DIVWAG Ground Combat Model. Each time an ADVANCE order is encountered within a Unit Scenario, the next command is checked for an ENGAGE order. If the ENGAGE order is not found, the ADVANCE is executed as a simple movement event. If an ENGAGE order is found, the following sequence of events takes place:

1. The list of units involved in the battle named in the ENGAGE order is obtained from the battle declaration statement of the appropriate Battle Paragraph.

2. From the list of all units involved in the battle, those units which are on the opposing side of the unit in whose scenario the ENGAGE order appears and whose current order is either PREPARE or WITHDRAW are selected. These units constitute potential opponents.

3. The list of potential opponents is checked for proximity to the advancing unit. If no potential opponents are within 3000 meters

(front-to-front) of the advancing unit, the battle is not initiated. (The process will be repeated as each Movement Model increment generated by the ADVANCE order is initiated.)

4. If a potential opponent is within 3000 meters of the advancing unit, the battle is initiated by scheduling the first Ground Combat Model increment. The increment is scheduled to take place after the standard Ground Combat Model increment length (15 minutes) has elapsed.

5. At the scheduled time, any unit listed in the Battle Paragraph which has an ADVANCE, PREPARE, or WITHDRAW order (a combat enabling order) will engage all units of the opposing force, also listed in the Battle Paragraph, and also having a combat enabling order and within 3000 meters.

6. The force to which the unit that had the initiating ENGAGE order belongs is treated as the attacking force within the Ground Combat Model.

(c) MISSION IS. The MISSION IS order is used to control the Air Ground Engagement Model. The order clause must contain the three mandatory modifiers TARGET NUMBER (a target intelligence index code from a DIVWAG intelligence report), NUMBER OF AIRCRAFT (a specified number), and AIRCRAFT TYPE (a specified equipment item code) and may contain the optional modifier AT TIME (a specified time). For example:

MISSION IS TARGET NUMBER 480001 AIRCRAFT TYPE 87  
NUMBER OF AIRCRAFT 4 AT TIME 030720.

This order must be within the Unit Scenario of a unit that is capable of aerial attack of a target. Upon receipt of the order a mission unit is formed from the unit receiving the order, where composition of the mission unit is as specified in the order; and the Air Ground Engagement Model simulation of the mission unit strike against the specified target is scheduled. If a time is specified, and it is possible to strike the target at the specified time, the Air Ground Engagement Model events are scheduled accordingly. If a time is not specified or is specified but unattainable, scheduling of the first portion of the model takes place at once; and the strike will occur after appropriate delays within the model data base have passed. To function properly, the unit must have the specified number and type of aircraft available (except if an Air Force unit and the aircraft is an appropriate TACAIR item). This may be ensured by use of the RETAIN order, discussed in a subsequent paragraph.

(4) Engineer Activity Orders. The DIVWAG Engineer Model functions as a resource allocation filter, controlling the assignment of engineer resources to tasks that must be accomplished. In consonance with this approach, DSL orders controlling engineer activity are task oriented rather than unit oriented. For a given game period, all commands requesting engineer tasks are consolidated within one Unit Scenario for each force, and the scenarios

given the unit identification BLUEFORC or REDFORCE as appropriate. Three basic DSL orders are used to initiate engineer activity, and one DSL order stops activity on a given task.

1. Activity Initiation Syntax. The basic DSL orders to initiate engineer activity are BUILD, BREACH, and REMOVE. Order clauses are built around these basic orders identically. The order clause contains the basic order; one of the exclusive modifiers BARRIER or BRIDGE or FACILITY, followed by a barrier facility identification; one of the exclusive modifiers BEGIN BY or COMPLETE BY, followed by a specified time; optionally, the modifier PRIORITY followed by a priority indicator 1, 2, 3 or 4; optionally, one of the modifiers MANDATORY or DESIRED. Several examples follow:

```
BUILD BARRIER MNA007 BEGIN BY 010800.  
REMOVE BARRIER MNA028 MANDATORY  
    COMPLETE BY 020705.  
BREACH MANDATORY PRIORITY 1 COMPLETE BY  
    010715 BARRIER MNT001.  
BUILD BRIDGE BRFO11 BEGIN BY 022200  
    DESIRED.  
REMOVE FACILITY BFL003 BEGIN BY 041730  
    PRIORITY 3.
```

2. Activity Initiation Response. The BUILD function is defined as construction of a new barrier or facility or repair and rebuilding of one that has been breached. The BREACH function is the disruption of the functional purpose of the obstacle or facility, while the REMOVE function is total destruction or dismantling and removal for use elsewhere. The modifiers BRIDGE and FACILITY are interchangeable. The Engineer Model operates upon a group of tasks, assigning resources to complete or initiate tasks as closely to the times requested as is possible. The user should be familiar with the priority algorithms used by the Engineer Model when assigning the optional modifiers. Briefly, all task orders having the modifier MANDATORY will have preference before task orders having the modifier DESIRED. If neither of the modifiers is used, the task order is treated as if the DESIRED modifier were present. The PRIORITY modifier and its data are used as a tie-breaker between task orders otherwise identical. If the modifier is not used, the task order is treated as though the modifier PRIORITY 4 had been present.

3. Relation to Automatic Task Requests. The DIVWAG Movement Model will generate automatic requests when a moving unit encounters an obstacle. The order will be to BREACH if the obstacle is breachable or to BUILD a facility if the obstacle is a natural obstacle. An automatic order will always be treated as having the modifier MANDATORY, the modifier BEGIN BY (time the obstacle is encountered), and PRIORITY (the priority associated with the movement event).

4. Task Cessation. Any engineer activity may be halted by the DSL order STOP TASK (barrier/facility identification). For example:

```
STOP TASK MNA013.
```

Work on the specified task halts at once, and any resources committed to the task are available for reassignment.

(5) Transfer Orders. The set of transfer activity orders provide a means for various administrative and organizational controls to be implemented within the model.

(a) JOIN. The JOIN order has one mandatory modifier UNIT (a UID). For example:

JOIN UNIT R1122333.

Receipt of this order causes the unit so ordered to integrate into another organization. The unit is completely absorbed by the receiving unit, its strength being reflected by the receiving unit in any subsequent actions. Once a unit joins another unit, it can accept no more orders until it has been detached. This order is used to change task organizations and, concurrently, results in the detail of resolution being reduced as several units join into one superior. If the receiving unit (unit being joined) had been a nonresolution unit prior to execution of the order it assumes the location of the joining unit and, if a Unit Scenario is provided, will immediately execute the first command of its Unit Scenario.

(b) DETACH. The DETACH order requires the mandatory modifier UNIT (a UID) and may have the optional modifier TYPE (a UTD). For example:

DETACH UNIT R1122BAR.  
DETACH UNIT B113MECH TYPE GRMI.

The DETACH order causes the named subunit to be broken out of the superior unit receiving the order (e.g., one company out of a battalion). Without the optional modifier, TYPE, the detached unit is formed with its pro rata share of the superior unit's strength (personnel and equipment). When the optional modifier, TYPE, is used, the detached unit will be broken out at its full authorized strength, to the extent that the strength of the superior unit allows. The detached unit is initially given the same location as the unit from which it was detached and will immediately begin to follow any DSL orders provided for it within a Unit Scenario. The detaching, or superior, unit will maintain all residual strength, if any, and will continue to follow DSL orders as long as it has residual strength. Combined use of the DETACH and JOIN orders provides the ability to restructure organizations in any manner desired, limited only by the quantity and definition of units within the game. Used in isolation, the DETACH order results in an increase in the detail of unit resolution. Proper functioning of the order requires that the unit being detached is already defined to the game, insofar that it must have been defined as a subordinate of the unit from which detached either within the task organization at the start of the game or through a previous JOIN order. The DETACH order with optional TYPE modifier may also be used to introduce new units to the game if, within the original task organization, the unit receiving the DETACH order was defined as being of a nonbasic UTD

and having no subordinates. In this case the UTD of the superior unit must be defined within TOE data as containing the UTD of the unit specified in the DETACH order.

(c) ASSUME CONTROL OF. This order, followed by the modifier, UNIT, causes the named unit to come under control of the unit receiving the order. Upon implementation of the order, status reports for the assuming unit will reflect strengths of the new subordinate. If the named unit is not already under control of the assuming unit and is not a resolution unit, it will be detached from its superior upon execution of the order. Example:

ASSUME CONTROL OF UNIT B1212123.

(d) ASSIGNMENT IS. This order, followed by an exclusive modifier, DIRECT SUPPORT OF UNIT, REINFORCING UNIT, GENERAL SUPPORT, or GENERAL SUPPORT-REINFORCING UNIT, causes the unit receiving the order to have the designated assignment for allocation of its supporting resources. Applications are within the TACFIRE Model, where fire support is allocated per assignment; within the Air Ground Engagement Model, where aerial fire support may be allocated by assignment; within the Airmobile Model, where allocation of lift and escort aircraft is per assignment; and in the Engineer Model, where allocation of engineer resources may be per assignment. Only the DIRECT SUPPORT assignment is actually acted upon within the Period Processor, the other assignments simply provide the user means of documenting organizational features of the force. In all cases, the GENERAL SUPPORT assignment serves to cancel previous assignments. Examples:

ASSIGNMENT IS GENERAL SUPPORT.  
ASSIGNMENT IS DIRECT SUPPORT OF UNIT B1111111.  
ASSIGNMENT IS REINFORCING UNIT B1234567.  
ASSIGNMENT IS GENERAL SUPPORT-REINFORCING  
UNIT B1212123.

(e) ACCEPT TRANSPORT. This order initiates the first segment of the Airmobile Model, which allocates aircraft and moves the aircraft to the location of the airmobile task force to be lifted. The ACCEPT TRANSPORT order must appear in the Unit Scenario of the unit being lifted and must precede the AIRMOBILE ASSAULT order for which aircraft are being allocated. The MIX modifier identifies the type lift and escort aircraft to be used and a standard lift to escort ratio. (This ratio is overridden by use of the NUMBER OF ESCORTS modifier.) The Airmobile Model allocates the specified number of lift aircraft, if the NUMBER OF AIRCRAFT modifier is used, or sufficient aircraft to move the entire unit in the specified number of trips, if the NUMBER OF TRIPS modifier is used. If the AT TIME modifier is used, aircraft are scheduled to arrive at the location of the unit to be lifted at the specified time. If such a time is not specified aircraft arrive as soon as possible after receipt of the order.

(f) RELEASE TRANSPORT. This order requires no modifiers or data. For example:

RELEASE TRANSPORT.



The RELEASE TRANSPORT order may appear in the Unit Scenario of an airmobile task force. If it appears it must not precede the AIRMOBILE ASSAULT order. In response to the RELEASE TRANSPORT order, all aircraft are returned to their home bases upon completion of the airmobile movement or upon receipt of the order, whichever occurs later. If the order is not used, aircraft remain with the airmobile task force.

(g) RETAIN. The RETAIN order has two mandatory modifiers: NUMBER OF AIRCRAFT (a specified number) and AIRCRAFT TYPE (an equipment item code). For example:

RETAIN AIRCRAFT TYPE 188 NUMBER OF AIRCRAFT 8.

This order should be reserved for use in the Unit Scenario of a unit which is capable of flying attack helicopter strikes within the Air Ground Engagement Model. When the order is used, the specified type and number of aircraft are exempt from automatic scheduling by the Air Ground Engagement Model. The order is intended to be used in conjunction with the MISSION IS order, to ensure availability of sufficient aircraft to conduct strikes ordered by the game staff. Aircraft are released for automatic scheduling as they return from a DSL ordered strike. Aircraft may also be released for automatic scheduling by a new RETAIN order, where the specified number is less than the number of aircraft currently retained. For example:

RETAIN AIRCRAFT TYPE 188 NUMBER OF AIRCRAFT 0.

(6) Pseudo Orders. Two DSL pseudo orders can be used in the construction of Unit Scenarios. The orders are so called because they differ from those described above with respect to their reasons for existence. They are not used to direct units to perform military activities but rather are inserted into Unit Scenarios to perform the functions described below.

(a) GO TO. The GO TO order is used to direct a unit to follow a set of orders in a particular sequence or with particular exceptions, depending on the order string structure and satisfaction of conditional statements. A GO TO order must be followed by a statement label. For example, the statement GO TO 6 may be inserted in an order string to direct the unit to start executing a statement labeled 6 immediately. Statement 6 may follow several others in the order string. If so, the orders between the statement GO TO 6 and statement 6 will be ignored by the unit.

(b) TERMINATE. This order, when encountered in any unit's scenario, will cause all simulation to halt and terminate the period. It is most useful in those cases where important or critical situations are expected to occur and for which contingency planning is not feasible or particularly desirable. The TERMINATE order should be used sparingly and with caution because it will halt all processing. To continue the war game a new game period, with complete DSL instructions for all units, must be prepared.

1 APR 1976

(7) Scatterable Mine Order. The emplacement of scatterable mines by indirect fire or aerial delivery means is directed by the EMLACE order. This order is limited to field emplacement by indirect fire or aerial means. The emplacement of scatterable minefields by more conventional ground means is treated through the Engineer BUILD order described in paragraph 3b(4) above.

(a) System. The EMLACE order must be followed by the mandatory modifiers FIELD (followed by a 6-character minefield identifier) and MUNITION TYPE (followed by a 4-character munition code). Additionally, one of the exclusive modifiers NUMBER OF ROUNDS, NUMBER OF VOLLEYS, or NUMBER OF TRIPS (followed by an integer number) must be used.

(b) Emplacement by Indirect Fire. Emplacement of minefields by indirect fire can be accomplished by, and the order may be given to, only artillery firing units. Such units are identified within the model as units which have the letters FA in the third and fourth characters of the Unit Type Designators. For example, units with the following UTD would accept the EMLACE order for indirect fire minefield delivery: GFFA, MBFA, NSFA, etc.

1. The field to be emplaced is identified by the modifier FIELD (field identification where the field identification is a six-character mnemonic. The first three characters must be MVA, MNP, MNT, or MNS and the last three characters must be integers in a range depending on the first three:

MVA001 - MNA500

MNP001 - MNP150

MNT001 - MNT150

MNS001 - MNS500

2. The munition mnemonic must be of the form AOnn where nn ranges from 01 to 36; this should be the code of a weapons/munition mix loaded for minefield delivery within the Area Fire Model data load.

3. In response to the EMLACE order, the ordered unit will fire the designated number of rounds or volleys of the designated type into the area of the specified minefield. Delivery is dependent on the field being within the range capabilities of the specific weapon/munition combination. Should the firing unit have fewer than the desired number of rounds, all munition available to the unit will be fired.

(c) Emplacement by Aerial Means. Aerial emplacement can only be accomplished by an aerial-type unit; that is, by a unit having a UTD which ends with the character H or Y. The same restrictions on the minefield name as were presented for indirect fire delivery apply.

III-2-A-17a

1 APR 1976

1. For aerial emplacement, the MUNITION TYPE modifier must be followed by a 4-character aerial emplacement code.

a. First character of the aerial emplacement code must be A, C, or H, designating emplacement by high performance (A), fixed wing cargo type (C), or helicopter (H). If helicopter delivery, the UTD of the unit receiving the order must end with H; otherwise, the UTD of the ordered unit must end with Y.

b. Second character of the aerial emplacement code must be an inter 0, 1, or 2. This specifies mission abort criterion: 0=do not abort in any case; 1=abort if aircraft fired upon; 2=abort if aircraft losses are experienced.

c. Third character of the aerial emplacement code is an integer in the range 1-9. This specifies an index for mean height at the dispensing site, where the height values are loaded in constant data base.

d. Fourth character of the emplacement code is an integer 1-9. This specifies an aircraft mix table (type and number of mines and type of aircraft) to be used in emplacement. The mix tables are part of the model's constant data load.

2. In response to the order, the number of aircraft specified in NUMBER OF TRIPS of the type in the appropriate mix table will be given full loads, as specified in the mix table, and will be sent to emplace the named field as a single mission unit.

c. DSL Order Modifiers and Data Elements. Most of the DSL order modifiers and some basic DSL orders require data elements to complete an order clause. The required format of a data element depends on the order or order modifier with which it is associated. This paragraph discusses the order modifiers and all required data elements.

(1) There are two basic rules regulating the entry of data elements when required by an order or an order modifier:

(a) A data element must follow the order or order modifier with which it is associated. In the following examples data elements are underlined:

FLY AT SPEED 150 KNOTS AT ALTITUDE 3000 FEET  
OVER 0112000-0098500, 0115000-0127500.

ADVANCE TO 0113000-0127000 BY TCCD  
AT WIDTH 2000M-DEPTH 1500M PRIORITY 1.

(b) Units of measure may be used as parts of data elements to provide clarification; however, the system requires that each type of data be input in a specific unit of measure. For example, aircraft speed must be in knots, and aircraft altitude must be in feet. Only units of time measure are actually recognized as key words by the compiler. Periods must not be used to terminate abbreviations; they are used exclusively to terminate statements. If units of measure are used in DSL statements, they must be restricted to the following forms:

- . DAY, DAYS, DA, or DAS
- . HOUR, HOURS, HR, or HRS
- . MINUTE, MINUTES, MIN, or MINS
- . FEET or FT
- . METER, METERS, or M
- . KNOT or KNOTS.

(2) Format of data elements generally must follow a rigid pattern. An exception is the specification of a time or a period of time. Times may be expressed either by a six-digit data time group or in clear text. In either case interpretation of the data element as a specific time or as a period of time depends on the modifier with which it is used.

(a) The first form uses an integer number of six digits with the digits in three blocks of two digits each. The blocks are in fixed order. The leftmost block represents the number of days, the center block the number

of hours, and the rightmost block the number of minutes. If fewer than six digits are used, a zero fill on the left is assumed. Examples:

112233 (denoting 11 days, 22 hours, and 33 minutes)

1122 (read as 001122 with zero fill on the left, denoting 11 hours and 22 minutes)

0800 (read as 000800, denoting 8 hours)

126 (read as 000126, denoting 1 hour and 26 minutes).

(b) The second form uses integer numbers of days, hours, and minutes, each of which must be followed by the word DAY, HOUR, or MINUTE (or abbreviations thereof as listed above). The words DAY, HOUR, and MINUTE (or abbreviations) are recognized by the compiler; therefore, the data may be written in any order and with any omissions. Examples:

1 DAY 12 HOURS 10 MINUTES

12 HOURS 10 MINUTES 1 DAY

12 HR 10 MIN

120 MIN

2 HRS

(3) For those orders and order modifiers that require data elements, specific data formats have been established. All order modifiers and those basic orders requiring data elements are presented below in alphabetical order.

(a) AIRCRAFT TYPE (equipment item code). The data element is an integer between 1 and 200 denoting the equipment item code of aircraft. Example:

AIRCRAFT TYPE 188

(b) AT ALTITUDE (height of aircraft in feet). The data element is an integer. Example:

AT ALTITUDE 6000 FT

(c) AT SPEED (velocity of aircraft in knots). The data element is an integer. Example:

AT SPEED 375

(d) AT TIME (a specified time). The data element is a time group as described above. Examples:

AT TIME 011230  
AT TIME 1 DAY 12 HR 30 MIN

(e) AT WIDTH (unit frontage in meters). The data element is an integer value used in conjunction with-DEPTH. Example:

AT WIDTH 2000M -DEPTH 1200M

(f) BARRIER (barrier identification code). The data element is a six-character mnemonic, the first three characters being alphabetic and the last three characters numeric. Example:

BARRIER MNA013

(g) BEGIN BY (a specified time). The data element is a time group as described above. Examples:

BEGIN BY 010800  
BEGIN BY 8 HOURS 1 DAY

(h) BRIDGE (facility identification code). The data element is a six-character mnemonic, the first three characters being alphabetic and the last three numeric. Example:

BRIDGE BFX103

(i) BY (code). Two cases exist:

1. BY (movement mode mnemonic). The data element consists of four alphabetic characters, comprising a movement mode mnemonic for the Movement Model. See Figure III-2-A-1. Example:

BY TCCD

2. BY (reconnaissance control code). The data element consists of four alphanumeric characters, the first of which is A, M, F or H, comprising a control code for the Reconnaissance Overlay. Example:

BY AXX3  
BY MAL6  
BY HR36  
BY H425

Movement type -

- A - Administrative. Movement of units by road nets. Uses the most efficient transportation systems available.
- T - Tactical. Movement as part of an attack, withdrawal, or other tactical plan external to movement within Ground Combat engagements.

Route type -

- CC - Cross country. Route is subject to natural terrain conditions.
- RA - Paved roads. Route is such that road beds are asphalt or concrete with at least two lanes with good shoulders.
- RG - Gravel roads. Route is gravel or similar surfaced road.
- RD - Dirt roads. Route is dirt, road is narrow and/or marginally maintained.

Formation -

- M - Column march. Unit is in a column formation.
- R - Reconnaissance. Unit on a ground reconnaissance type mission.
- D - Deployed. Unit is partially deployed in anticipation of imminent contact with the enemy.

Recognized movement combinations -

A:	RAD	RDD	RGD	
	RAM	RDM	RGM	
	RAR	RDR	RGR	
T:	RAD	RDD	RGD	CCD
	RAM	RDM	RGM	CCM
	RAR	RDR	RGR	CCR

Figure III-2-A-1. Travel Mode Mnemonic Description

(j) COMPLETE BY (a specified time). The data element is a time group as described above. Examples:

COMPLETE BY 031720  
COMPLETE BY 3 DAYS 17 HR 20 MIN

(k) -DEPTH (unit depth in meters). The data element is an integer value used in conjunction with AT WIDTH. Example:

AT WIDTH 1250 -DEPTH 1525

(l) DESIRED. No data element.

(m) DIRECT SUPPORT OF UNIT (UID of supported unit). The data element is an eight-character alphanumeric unit identification beginning with B or R. Example:

DIRECT SUPPORT OF UNIT B1212AAR

(n) FACILITY (facility identification code). The data element is a six-character mnemonic, the first three characters being alphabetic and the last three numeric. Example:

FACILITY BFX103

(o) FOR (a specified period of time). The data element is a time group as defined above. The following examples are equivalent:

FOR 200  
FOR 2 HOURS  
FOR 120 MIN

(p) GENERAL SUPPORT. No data required.

(q) GENERAL SUPPORT-REINFORCING UNIT (UID of the reinforced unit). The data element is an eight-character alphanumeric unit identification beginning with B or R. Example:

GENERAL SUPPORT-REINFORCING UNIT R1234TKB

(r) GO TO (label of procedure statement). GO TO is a pseudo order used to direct the sequence of unit procedure statements applicable to a unit. The data element is an alphanumeric character string of one to three characters. In the following example, the labeled statement is also shown.

GO TO A12.  
A12: STAY FOR 3 HRS.



(s) HEIGHT OF BURST (height index). The data element is an integer value 0, 1, 2, 3, 4. Zero is used for a nuclear fire event in which height of burst of the munition and fuze may be specified. Where munition and fuze permit only preset heights of burst, the integer 1, 2, 3, 4 is an index to the preset height option desired, as defined through Nuclear Assessment Model constant data. Example:

HEIGHT OF BURST 2

(t) IMPACT RADIUS (desired height of nuclear burst in meters). The data element is an integer. It is acted upon only in the case of a nuclear munition and fuze which allows specification of height of burst, in which case the data element is the desired height of burst in meters. Example:

IMPACT RADIUS 500M

(u) IN BATTLE (battle identification). The data element is composed of no more than eight alphanumeric characters. Examples:

IN BATTLE EIGHTCHR  
IN BATTLE X  
IN BATTLE 1PLUS2

(v) MANDATORY. No data required.

(w) MIX (an airmobile aircraft mix index). The data element is an integer. Example:

MIX 3

(x) MUNITION TYPE (weapon/munition code). The data element is a four-character code, the first character being A (conventional round), N (nuclear round), or D (atomic demolition munition). Examples:

A012  
NKX3

(y) NUMBER OF AIRCRAFT (specified number of aircraft). The data element is an integer. Example:

NUMBER OF AIRCRAFT 4

(z) NUMBER OF ESCORTS (specified number of escort aircraft). The data element is an integer. Example:

NUMBER OF ESCORTS 6

(aa) NUMBER OF ROUNDS (specified number). The data element is an integer. Example:

NUMBER OF ROUNDS 187

(bb) NUMBER OF TRIPS (specified number of airmobile lift trips). The data element is an integer. Example:

NUMBER OF TRIPS 2

(cc) NUMBER OF VOLLEYS (specified number). The data element is an integer. Example:

NUMBER OF VOLLEYS 1

(dd) ON (location). The data element is a single rectangular map coordinate entry of the form integer - integer. Each integer can be one to seven digits in length. If fewer than seven digits are used, leading zeros are assumed by the compiler. Example:

ON 1163590 - 1246780

(ee) OVER (location or list of locations). A map coordinate pair or list of as many as eight pairs may be entered. Each pair has the form integer - integer, where each integer can be one to seven digits in length; leading zeros are assumed if fewer than seven digits are used. Each location is to the nearest meter. Examples:

OVER 163590 - 246780

OVER 163590 - 246780, 163600 - 246800

(ff) PRIORITY (movement or engineer priority). The data element is one of the integer values 1, 2, 3 or 4. Example:

PRIORITY 4

(gg) REINFORCING UNIT (UID of reinforced unit). The data element is composed of eight alphanumeric characters, the first of which must be B or R. Example:

REINFORCING UNIT B1212123

(hh) STOP TASK (barrier or facility identification). The data element consists of three alphabetic and three numeric characters. Example:

STOP TASK MNA017

(ii) TARGET NUMBER (target index from intelligence report). The data element is an integer. Example:

TARGET NUMBER 480003

(jj) TO (location or list of locations). A map coordinate pair or list of pairs may be entered. Each pair has the form integer - integer, where each integer can be one to seven digits in length; leading zeros are assumed if fewer than seven digits are used. Each location is to the nearest meter. Examples:

TO 163590 - 246780  
TO 163590 - 246780, 163600 - 246800

(kk) TYPE (a unit type designator). The data element is composed of four alphabetic characters. Example:

TYPE BAMT

(ll) UNIT (a unit identification). The data element consists of eight alphanumeric characters, the first of which is B or R. Example:

UNIT B1212123

(mm) UNTIL (a specified time). The data element is a time group as described above. Examples:

UNTIL 021215  
UNTIL 02 DAYS 12 HOURS 15 MINUTES

#### 4. DSL CONDITIONAL CLAUSES:

a. General. Conditional clauses are used to specify the user's desired sequence of order execution when that sequence depends on conditions prevailing prior to execution of an order or during or upon termination of an order currently being executed. Conditional clauses can be used in Unit Scenarios and in Battle Paragraphs. As with order clauses, the construction of a conditional clause is governed by the allowed vocabulary and the DSL syntax.

b. Conditional Clause Vocabulary. A limited vocabulary is used in the construction of conditional clauses. This vocabulary, together with four possible types of data entries, constitutes the totality of symbols recognized by the DSL Compiler in processing conditional clauses. The vocabulary elements can be arranged within 14 distinct groups, of which groups 1, 12, 13, and 14 are data entries:

<u>Group Number</u>	<u>Description</u>	<u>Legal Vocabulary</u>
1	Unit identification	A UID consisting of eight alpha-numeric characters, the first of which is B or R.
2	Unit's possession	CLASS 3, CLASS 5, PRESENT STRENGTH
3	Unit's possession with data	EQUIPMENT TYPE XXX (an equipment item code)
4	Clock time	TIME
5	Logical operator	GREATER THAN, LESS THAN, EQUAL TO
6	Negator	NOT
7	Unit's activity	ASSESSED, FIRING, MOVING, STOPPED
8	Weather condition	CLOUD COVER, FOG INDEX, RELATIVE HUMIDITY, TEMPERATURE, TEMPERATURE GRADIENT, VISIBILITY INDEX, WIND DIRECTION, WIND SPEED, PRECIPITATION INDEX
9	Location of weather condition	AT LOCATION
10	Unit's activity with data	HALTED AT, AT LOCATION
11	Percent indicator	PERCENT
12	Location data	A pair of seven-digit map coordinates; e.g., 0122000-0095750.
13	Quantity data	An integer.
14	Time data	Formats as presented in subparagraph 3c(2).

c. Conditional Clause Syntax. Syntax established for the DSL Compiler allows seven conditional clause types, where a clause type is a set sequence of elements from the conditional clause vocabulary groups. The types are listed below with one example of each type. Parentheses indicate a group is optional.

<u>Type</u>	<u>Pattern</u>	<u>Example</u>
A	1 2 (6) 5 13 (11)	B1234567 PRESENT STRENGTH GREATER THAN 150
B	1 3 (6) 5 13	B1234567 EQUIPMENT TYPE 172 LESS THAN 34
C	1 8 (6) 5 13 (11)	B1234567 VISIBILITY INDEX NOT LESS THAN 5
D	8 9 12 (6) 5 13 (11)	CLOUD COVER AT LOCATION 0123500-0095000 NOT GREATER THAN 70 PERCENT
E	4 (6) 5 14	TIME GREATER THAN 011300
F	1 (6) 7	B1234567 NOT MOVING
G	1 (6) 10 12	B1234567 AT LOCATION 0123000-0095800

d. Description of Conditions being Tested:

(1) Conditional Type A. The unit for which the test is made may be any resolution unit defined within the system. The quantity against which a check is made is the quantity present in the specified unit of personnel (PRESENT STRENGTH); Class 3, or equipment item 2 (CLASS 3); ammunition associated with individual weapons, or equipment item 6 (CLASS 5). If PERCENT is used, the quantity against which a check is made is the ratio of the amount present in the unit to amount authorized. Examples:

B1234567 CLASS 3 NOT LESS THAN 2500  
B1234567 CLASS 5 GREATER THAN 70 PERCENT  
B1234567 PRESENT STRENGTH LESS THAN 150

(2) Conditional Type B. The unit for which the test is made may be any resolution unit defined within the system. The item on which the check is to be made is specified by an equipment item code between 1 and 200. The quantity against which a check is made is the quantity present in the unit or, if PERCENT is used, the ratio of quantity present to authorized quantity. Example:

B1234567 EQUIPMENT TYPE 32 LESS THAN 500  
B1234567 EQUIPMENT TYPE 3 LESS THAN 50 PERCENT

(3) Conditional Type C. The value against which a check is made is the current weather condition at the location of the specified unit. Legal values of the data entry depend on the weather condition being checked. The values which are treated as percentages are so indicated.

(a) Cloud Cover. The data value can be an integer from 1 to 100. It is treated as a percentage. Example:

B1234567 CLOUD COVER LESS THAN 50 PERCENT

(b) Fog Index. The data value can be 0 (no fog) or 1 (fog). PERCENT is not allowed. Limitation to the logical operators EQUAL TO and NOT EQUAL TO is suggested. Example:

B1234567 FOG INDEX EQUAL TO 1

(c) Relative Humidity. The data value must be between 1 and 100. It is treated as a percentage. Example:

B1234567 RELATIVE HUMIDITY NOT LESS THAN 60 PERCENT

(d) Temperature. The data entry should be desired temperature in degrees Fahrenheit. Example:

B1234567 TEMPERATURE LESS THAN 32

(e) Temperature Gradient. The data value may be 1 (inversion), 2 (moderate inversion), 3 (neutral), or 4 (lapse). Example:

B1234567 TEMPERATURE GRADIENT LESS THAN 3

(f) Visibility Index. The data entry may range from 1 to 9 where 9 denotes best and 1 denotes poorest visibility. Example:

B1234567 VISIBILITY INDEX NOT LESS THAN 6

(g) Wind Direction. The data value, denoting azimuth in degrees, may range from 0 to 360. Example:

B1234567 WIND DIRECTION NOT LESS THAN 45

(h) Wind Speed. The data entry is wind speed in knots. Example:

B1234567 WIND SPEED NOT LESS THAN 12

(i) Precipitation Index. The data entry may have a value of 0 (no precipitation), 1 (light precipitation), or 2 (heavy precipitation). Example:

B1234567 PRECIPITATION INDEX EQUAL TO 0

(4) Conditional Type D. The value against which a check is made is the current weather condition at the specified location. Rules are identical to those for conditional type C.

(5) Conditional Type E. The value within the clause is any legal time format as described in subparagraph 3c(2). This is checked against current time. Example:

TIME GREATER THAN 011330

(6) Conditional Type F. The condition checked is the current status of the specified unit as follows:

(a) ASSESSED. A unit is sensed as ASSESSED if it has been attrited by the Area Fire or Air Ground Engagement Models within the past 15 minutes. Example:

B1234567 NOT ASSESSED

(b) FIRING. A unit is sensed as FIRING if it has received a DSL FIRE order or a TACFIRE fire mission and has not yet completed the ordered fire mission. Example:

B1234567 FIRING

(c) MOVING. A unit is sensed as MOVING if it has received one of the following orders and has not reached the final movement coordinates: MOVE, FLY, ADVANCE, WITHDRAW. If, however, the order is ADVANCE or WITHDRAW and the unit has actually engaged in ground combat, it is not sensed as moving. Example:

B1234567 MOVING

(d) STOPPED. The STOPPED condition is exactly equivalent to NOT MOVING, and MOVING is exactly equivalent to NOT STOPPED.

(7) Conditional Type G. The condition checked is whether the specified location is within the rectangular area defined by the specified unit's current location, orientation, width, and depth. The specified unit must be a resolution unit. Example:

B1234567 NOT AT LOCATION 0123000-0095000.

5. CONTROL CARDS AND DECK STRUCTURE. This paragraph presents control cards required by the DSL Compiler, structure of the DSL Compiler data deck, and structure of decks for submittal to the data processing facility.

a. Compiler Control Cards:

(1) The DSL Compiler call card must be the first card of the data deck. This card has one of the following forms:

DSL.

DSL, DEBUG.

Position on the punched card is not critical, as any blanks are ignored; however, the first entry must be DSL, and the designated comma and period must appear as indicated. Use of the DEBUG option causes the tables generated by the DSL compiler to be listed.

(2) The start of period card must be the second card of the data deck. This card has the form:

START OF PERIOD: XX DAY XX HOUR XX MINUTE.

where XX represents an appropriate numeric entry such as:

START OF PERIOD: 01 DAY 18 HOUR 30 MINUTE.

Presence of the colon and period as illustrated is crucial. To indicate the start of a game, the card must appear as follows:

START OF PERIOD: 01 DAY 00 HOUR 00 MINUTE.

This card indicates the game time at which the period for which DSL orders are being supplied is to start.

(3) The period length card must be the third card of the data deck. This card must be of the form:

PERIOD LENGTH: XXXX MINUTES.

where XXXX represents an appropriate numeric entry such as:



PERIOD LENGTH: 480 MINUTES.

The colon and period must appear as indicated. This card indicates the length of the period of combat to be simulated using the DSL orders.

(4) The final card of the DSL data deck must contain, in any position on the card, the notation:

FINIS.

This card indicates the end of the DSL data deck.

(5) Comments may be inserted at any position in the data deck following the third card. The comment is used to provide background elaboration of the operations being ordered for the information of any individual who should have access to the DSL data deck or a listing thereof. Comments have no meaning to the DSL Compiler or to any other processor of the DSL system and exist solely for user convenience. A comment must be introduced by one of the phrases COMMENT:, CONCEPT:, or CONCEPT OF OPERATION:, where the colon is an integral part of the phrase. A comment is ended with a period. The body of a comment, between the introductory phrase and closing period, may contain any symbol except a period. One comment may be of any length up to (and including) 400 nonblank characters. A DSL data deck may contain as many comments as desired. Example:

COMMENT: SECOND BRIGADE SHOULD NOW BE IN ATTACK  
POSITIONS; 1ST BN ON NORTH, 2ND BN ON  
SOUTH AND 3RD BN IN RESERVE (EAST) PD  
SUPPORTING ARTILLERY (1ST AND 3RD OF 307TH)  
HAVE FIRED PREPARATORY MISSIONS AND GONE  
TO TACFIRE MODE.

b. Data Deck Structure. The DSL data deck contains four segments as illustrated in Figure III-2-A-2.

(1) The first segment of a DSL data deck must comprise the compiler call card, start of period card, and period length card, in that sequence.

(2) Unit Scenarios comprise the second segment of a DSL data deck. The order in which individual Unit Scenarios appear is not critical, although review may be facilitated by keeping scenarios for units of the Red force and Blue force separate. A Unit Scenario is acted upon only if the specified unit is a resolution unit or has a personnel strength of at least one. Scenarios for nonresolution units are ignored, unless the unit should attain resolution status as the result of an appropriate transfer activity order (DETACH) within the scenario of another unit. In this case, execution of the scenario commences when resolution status is attained. The exception is a scenario with identification ID:REDFORCE. or ID:BLUEFORC, which contains all engineer activity orders for the appropriate force. Comments may appear within any Unit Scenario.

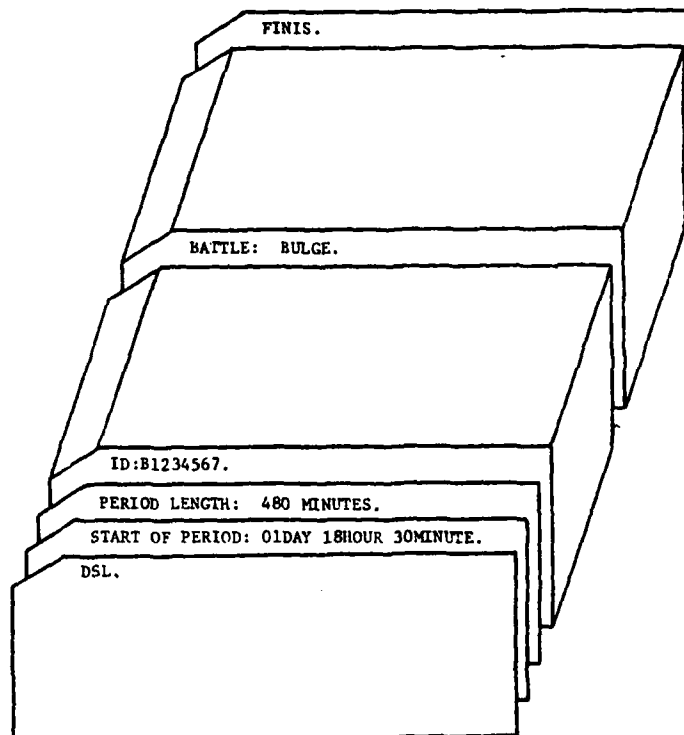


Figure III-2-A-2. DSL Data Deck Structure

(3) Battle Paragraphs comprise the third segment of a DSL data deck. All units listed in battle declaration cards of Battle Paragraphs must be provided Unit Scenarios which contain appropriately labeled commands keyed to the battle conditionals as discussed in subparagraph 2c. Situations may arise where a DSL data deck having no Battle Paragraphs is appropriate, which is allowed by the compiler.

(4) The final segment of a DSL data deck is the FINIS. card.

6. DSL RULES AND TECHNIQUES. The DIVWAG Scenario Language provides the gamer with a wide degree of latitude in controlling simulated units within the DIVWAG system. Proper use of DSL does, however, depend upon observation of the basic vocabulary and syntax rules of the language. Effective use also depends upon an appreciation of the model's response to the various orders. This paragraph restates some of the more critical basic DSL rules and provides selected guidelines toward effective DSL writing techniques. As with any language, an individual's facility with DSL strongly depends upon the extent of his exposure to and experience with the language. Thus, fluency with DSL cannot be expected simply by exposure to this manual. It is gained through practice.

a. Elementary Rules:

(1) Unit Scenarios:

(a) A Unit Scenario can be acted upon only if the unit is a resolution unit and has personnel. The DSL Compiler will accept Unit Scenarios for any unit, as long as the scenario is identified with a legal UID. The DIVWAG Period Processor, however, will ignore orders for units not defined within the game, for nonresolution units, and for resolution units having no personnel.

(b) A Unit Scenario may be provided for a nonresolution unit and that unit may gain a resolution status through appropriate use of the DETACH order given to another unit. When this happens, the Unit Scenario is acted upon at the time resolution status is gained.

(c) If a unit loses resolution status, through the JOIN order in its own Unit Scenario or through an order to DETACH its last subordinate; or should a unit lose all personnel; it will not execute any remaining orders in its Unit Scenario.

(d) A unit may have only one Unit Scenario. Presence of more than one Unit Scenario for a given unit causes the tables generated by the DSL Compiler to be invalidated.

(e) A Unit Scenario must contain at least one command.

(f) Liberal use of labels within a Unit Scenario is a key to efficient use of DSL. Labels must, however, be unique within a Unit Scenario. The sequence of orders cannot be followed as the user intends if more than one command has the same label in one Unit Scenario.

(2) Battle Paragraph/Unit Scenario Interface:

(a) Each unit listed in the battle declaration card must be provided a Unit Scenario.

(b) Within the Unit Scenario for each unit listed in the battle declaration card, properly labeled commands must appear.

b. Basic DSL Techniques:

(1) Timing. It is generally possible to control the time at which execution of an order begins by preceding the order with a STAY order or a PREPARE order. In the following example, the STAY order is used to time a FIRE order and a MOVE order.

ID:B1111111.  
STAY UNTIL 010530.  
FIRE MUNITION TYPE A003 ON 0123000-0123000  
NUMBER OF VOLLEYS 4 IMPACT RADIUS 100.  
STAY UNTIL 010615.  
MOVE TO 0118000-0127000.

(2) Repetitive Orders. It is generally possible to repeat a sequence of orders using the GO TO order with appropriate labels. In the following example, an artillery unit is ordered to fire upon a series of three targets repetitively. A time conditional is used to exit the loop when time exceeds 021800.

ID:B11111FA.  
STAY UNTIL 021630.  
A1:FIRE MUNITION TYPE A003 ON 0123000-0123000  
NUMBER OF VOLLEYS 3 IMPACT RADIUS 100.  
FIRE ON 0118000-0123000 MUNITION TYPE A003  
NUMBER OF VOLLEYS 3 IMPACT RADIUS 100.  
FIRE ON 0120000-0122525 NUMBER OF VOLLEYS 2  
MUNITION TYPE A003 IMPACT RADIUS 100.  
IF TIME NOT GREATER THAN 021800, THEN GO TO A1.  
STAY UNTIL 022000.

(3) Skipping Commands. Judicious use of conditionals with the GO TO order permits skipping a command or series of commands as the situation may dictate. In the example B123BN01 will engage in one of two battles depending upon the strength of B123BN02. The command with label ABC will be executed upon termination of either battle (Battle Paragraphs not shown here).

ID:B123BN01.  
PREPARE UNTIL 020500.  
IF B123BN02 PRESENT STRENGTH LESS THAN 125, THEN  
GO TO AAA.  
ADVANCE TO 0125000-0113000.  
ENGAGE IN BATTLE BIGHORN.  
AAA:ADVANCE TO 0120000-0115500.  
ENGAGE IN BATTLE FIREFITE.  
ABC:PREPARE UNTIL 021200.

c. Techniques Related to Models:

(1) Ground Combat Model. Preparation of DSL commands for the Ground Combat Model is the most subtle phase of DSL preparation. Functioning of the model must be held in mind as orders are prepared.

(a) Battle is initiated in response to an ADVANCE order followed by an ENGAGE order. Initiation requires scheduling the first Ground Combat Model assessment cycle to take place 15 minutes from the time of initiation. Initiation takes place only if a unit on the other force is currently under a PREPARE or a WITHDRAW order and is within 3000 meters (front to front) of the unit receiving the ENGAGE order.

(b) Assessment of battle results takes place at the scheduled time. Assessment will cover all units under ADVANCE, PREPARE, or WITHDRAW orders at the time of assessment. Only units under an appropriate order and within 3000 meters of an opponent who is also under an appropriate order are treated. Specification of attacker and defender within the Ground Combat Model is determined by the one unit in whose scenario the ENGAGE order that caused initiation occurs. All units on the initiator's side are assessed as attackers and all units on the opposing side are assessed as defenders.

(c) Upon completion of the assessment cycle, all conditionals in the Battle Paragraph are checked to determine if the battle must be terminated. If none of the conditions are met, another assessment cycle is scheduled to take place in 15 minutes.

(d) A battle may be reinitiated at any time another ENGAGE order is encountered. Once initiated and terminated, a battle will generally terminate 15 minutes after reinitiated. This is because reinitiation will schedule a 15-minute cycle, and the conditional that terminated the battle the first time will generally continue to be true and will generally terminate the battle each time it is reinitiated.

(e) Upon battle termination, all units listed in the Battle Paragraph will progress to labeled commands as indicated by the battle conditional that is met. This occurs regardless of whether the units actually participated in the battle. The command under execution at the time of battle termination is not generally completed.

(f) Battle conditionals are checked in the order in which they appear in the Battle Paragraph. Once a condition is met, other conditionals are not checked. Thus, if several conditions are included, the condition expected to occur at the latest time should generally appear first in the Battle Paragraph.

(2) Reconnaissance Orders. The reconnaissance control code is of paramount importance in a RECONNOITER order. The code consists of four characters,  $C_1C_2C_3C_4$ , where the first character,  $C_1$ , identifies the mission type as light observation helicopter mission ( $C_1=H$ ), light fixed wing aircraft mission ( $C_1=F$ ), Mohawk OV-1D type aircraft mission ( $C_1=M$ ), or Air Force reconnaissance mission ( $C_1=A$ ). Meaning of successive characters of the code depends upon the first character.

(a) Control Code Determination:

1. LOH/Fixed Wing Observation Reconnaissance Mission. Code equals  $H C_2C_3C_4$  or  $F C_2C_3C_4$ .

a. If the second character,  $C_2$ , is R, then the mission is a route reconnaissance mission; otherwise, it is an area reconnaissance mission. If it is an area reconnaissance mission, then  $C_2$  has values 1-9, where the integer value specifies the time assigned to the search area in units of 15-minute intervals. For example, a code of H326 specifies an area reconnaissance mission lasting 45 minutes for an LOH.

b. The third character,  $C_3$ , has a range of 0-9 and specifies the route deviation limit in kilometers (corridor width) which the aircraft will not exceed during the flight. In the example, H326, the character 2 in the third position specifies that the LOH will reconnoiter along routes with a corridor width of 2 kilometers, and thus will never exceed the 1-kilometer deviation limit from the route interval. For an area reconnaissance mission, the third character is used in the same manner and effectively creates a density of reconnaissance coverage; i.e., successive passes over the assigned area will be separated by the corridor width.

c. The fourth character,  $C_4$ , ranges from 1-9 and is used for two purposes:

- .  $C_4$  is the sensor load combination code which identifies the list of sensor types carried on board the aircraft
- .  $C_4$  is also the combination code which identifies the correct LOH decision control matrix to use in this mission. Note that the same sensor load is required to use the same decision matrix; however, with up to 10 combinations available, it is possible to list the same sensor load with different decision matrices.

2. Mohawk Type Mission. Code equals M C<sub>2</sub>C<sub>3</sub>C<sub>4</sub>.

a. This type reconnaissance mission is currently restricted to represent only the SLAR MTI with GST and camera sensor types on board. The fourth character, C<sub>4</sub>, identifies the sensor load combination carried on board in the same manner as the LOH. The first sensor type in the combination is currently restricted to the SLAR MTI with GST.

b. The second character, C<sub>2</sub>, is used to set the range and delay of the SLAR MTI sensor package during the RECONNOITER order flight path. The allowed values of C<sub>2</sub> and interpretations are shown in Figure III-2-A-3 where the values of DELAY, RANGE 1, RANGE 2, and RANGE 3 are set as part of the constant data base.

c. The third character, C<sub>3</sub>, defines the direction(s) for which the SLAR is gated as being to the right, left, or to both sides of the aircraft flight direction as follows:

- . C<sub>3</sub> = R, radar is gated on the right side only
- . C<sub>3</sub> = L, radar is gated on the left side only
- . C<sub>3</sub> = B, radar is gated on both sides

3. Air Force Aircraft Reconnaissance Mission. Code equals A X X C<sub>4</sub>.

a. The second and third characters of the code are not currently used by the submodel and should contain X X.

b. The fourth character, C<sub>4</sub>, is used to specify the sensor load combination on board as in previous mission types. (Sensor types are currently restricted to camera systems.)

(b) DSL Flight Pattern Data:

1. The DSL order also specifies the flight intervals or area over which the reconnaissance mission is to be flown. The coordinate endpoints listed on the DSL order form the actual flight path taken in Mohawk OV-1D and Air Force reconnaissance missions.

2. If the mission is an area reconnaissance mission, the coordinates specify the four corners of the area over which the reconnaissance mission is to be flown. The order of the points appearing in the DSL order is such that P<sub>1</sub>P<sub>2</sub>P<sub>3</sub>P<sub>4</sub> are in counterclockwise order around the enclosed reconnaissance area. Also P<sub>1</sub>P<sub>2</sub> is the rear boundary from which the reconnaissance aircraft will start the coverage of the area.

3. If the DSL order request is for an LOH type mission, the actual flight path does not follow the route intervals exactly but remains within the corridor limits defined in the DSL order control code.

Second Character of code	SLAR MTI Settings	
	Delay Setting	Range Setting
C <sub>2</sub> = 0	0 x DELAY	RANGE1
C <sub>2</sub> = 1	1 x "	"
C <sub>2</sub> = 2	2 x "	"
C <sub>2</sub> = 3	3 x "	"
C <sub>2</sub> = 4	4 x "	"
C <sub>2</sub> = 5	5 x "	"
C <sub>2</sub> = 6	6 x "	"
C <sub>2</sub> = A	0 x DELAY	RANGE2
C <sub>2</sub> = B	1 x "	"
C <sub>2</sub> = C	2 x "	"
C <sub>2</sub> = D	3 x "	"
C <sub>2</sub> = E	4 x "	"
C <sub>2</sub> = F	5 x "	"
C <sub>2</sub> = Z	0 x DELAY	RANGE3

Figure III-2-A-3. Delay and Range Settings for Mohawk Type Mission

7. DIVWAG SCENARIO LANGUAGE VOCABULARY. Figure III-2-A-4 and Figure III-2-A-5 provide a compendium of the DSL order and modifier vocabulary in tabular form.

a. DSL Orders. Figure III-2-A-4 lists the DSL order, the appropriate modifiers--required, exclusive, and optional--for that order, and the code number.

b. DSL Order Modifiers. Figure III-2-A-5 lists the modifiers of DSL orders with the type and format of data. Various comments are included regarding the data format.



Orders	Required Modifiers	Exclusive Modifiers*	Optional Modifiers	Code Number
ACCEPT TRANSPORT	MIX	NUMBER OF TRIPS NUMBER OF AIRCRAFT	NUMBER OF ESCORTS AT TIME	20
ADVANCE	TO		BY PRIORITY AT WIDTH - DEPTH	6
AIRMOBILE ASSAULT	TO		AT TIME	41
ASSIGNMENT IS		DIRECT SUPPORT OF UNIT REINFORCING UNIT GENERAL SUPPORT GENERAL SUPPORT REINFORCING UNIT		23
ASSUME CONTROL OF UNIT				24
BREACH		BARRIER BRIDGE FACILITY BEGIN BY COMPLETE BY	PRIORITY MANDATORY } one of DESIRED } one of	43
BUILD		BARRIER BRIDGE FACILITY BEGIN BY COMPLETE BY	PRIORITY MANDATORY } one of DESIRED } one of	42

Figure III-2-A-4. DSL Orders (Continued on Next Page)

Orders	Required Modifiers	Exclusive Modifiers*	Optional Modifiers	Code Number
DETACH	UNIT		TYPE	26
ENGAGE	IN BATTLE			15
FIRE (Conventional)	MUNITION TYPE ON	NUMBER OF ROUNDS NUMBER OF VOLLEYS	IMPACT RADIUS	9
FIRE (Nuclear)	IMPACT RADIUS MUNITION TYPE NUMBER OF ROUNDS 1 ON		HEIGHT OF BURST	9
FIRE ON TARGETS OF OPPORTUNITY		FOR UNTIL		10
FLY	AT ALTITUDE AT SPEED OVER			7
GO TO	(command label)			
JOIN	UNIT			25
LOITER	AT ALTITUDE AT SPEED FOR			32
MISSION IS	AIRCRAFT TYPE NUMBER OF AIRCRAFT TARGET NUMBER		AT TIME	39
MOVE	TO		BY PRIORITY	3

III-2-A-40

Figure III-2-A-4. DSL Orders (Continued)

Orders	Required Modifiers	Exclusive Modifiers*	Optional Modifiers	Code Number
PREPARE		FOR UNTIL	AT WIDTH - DEPTH	2
RECONNOITER	AT ALTITUDE AT SPEED BY OVER			8
RELEASE TRANSPORT				31
REMOVE		BARRIER BRIDGE FACILITY BEGIN BY COMPLETE BY	DESIRE MANDATORY } one of	44
RETAIN	AIRCRAFT TYPE NUMBER OF AIRCRAFT			40
STAY		FOR UNTIL		1
STOP TASK	(barrier/facility identifier)			45
TERMINATE				37
WITHDRAW	TO		BY AT WIDTH - DEPTH PRIORITY	5

\* One and only one required.

Figure III-2-A-4. DSL Orders (Concluded)

Modifier	Data Type	Data Format	Comments
AIRCRAFT TYPE	Equipment item code	One to three digits	
AT ALTITUDE	Height (in feet)	Integer	Integer may be followed by FEET or FT.
AT SPEED	Speed (in knots)	Integer	Integer may be followed by KNOT or KNOTS.
AT TIME	Absolute time	ddhhmm  dDhHmM	<p>d = number of days (digit). h = number of hours (digit). m = number of minutes (digit). One to six digit integer. If less than six digit, zero fill on left is assumed.</p> <p>d = number of days (integer). h = number of hours (integer). m = number of minutes (integer). D = DAYS, DAY, or DA H = HOURS, HOUR, or HR M = MINUTES, MINUTE, or MIN</p> <p>dD, hH, or mM (or any two of the three forms) may be omitted (e.g., dDhH is valid). dD, hH, and mM may be arranged in any order (e.g., hHdD is valid). Caution: 0800 HOURS = 800 hours, not eight hours.</p>

Figure III-2-A-5. DSL Order Modifiers (Continued on Next Page)

Modifier	Data Type	Data Format	Comments
AT WIDTH	Unit front (meters)	Integer	
BARRIER	Barrier identification code	Alphanumeric string, six characters.	First three characters are alpha, last three numeric.
BEGIN 3Y	Absolute time	(See AT TIME modifier)	
BRIDGE	Facility identification code	Six alphanumeric characters.	First three characters are alpha, last three numeric.
BY	Transportation mode	Four alphabetic characters.	
BY	Reconnaissance control code.	Four alphanumeric characters.	Used only with RECONNOITER order. First character is A, F, L or M.
COMPLETE BY	Absolute time	(See AT TIME modifier)	
- DEPTH	Unit depth (meters)	Integer	
DESIRED	Not applicable.		
DIRECT SUPPORT OF UNIT	UID	Alphanumeric string	Eight characters beginning with B or R.
FACILITY	Facility identification code	Six alphanumeric characters.	First three characters are alpha, last three numeric.
FOR	Time length	(See AT TIME modifier)	
GENERAL SUPPORT	Not applicable		Used with ASSIGNMENT IS order.

Figur II-2-A-5. DSL der Modifiers (continued)

Modifier	Data Type	Data Format	Comments
GENERAL SUPPORT- REINFORCING UNIT	UID	Alphanumeric string	Eight characters beginning with B or R.
GO TO	Label	Alphanumeric string	Three characters only. One of two orders (not modifiers) requiring data.
HEIGHT OF BURST	Preset HOB option code.	Integer, 0, 1, 2, 3, 4	Used with nuclear FIRE order to indicate which preset HOB when fuzing appropriate.
IMPACT RADIUS	Height of burst (meters)	Integer	Used with nuclear FIRE when fuzing allows specification of HOB. Mandatory for all other FIRE orders but data not used.
IN BATTLE	Battle name	Alphanumeric string	One to eight characters.
MANDATORY	Not applicable.		
MIX	Airmobile transport/escort mix code.	Integer 1-10	
MUNITION TYPE	Munition mnemonic	Alphanumeric string	Four characters.
NUMBER OF AIRCRAFT	Desired number	Integer	Used with RETAIN, MISSION IS, ACCEPT TRANSPORT orders.
NUMBER OF ESCORTS	Desired number	Integer	Used with ACCEPT TRANSPORT order.
NUMBER OF ROUNDS	Rounds	Integer	For nuclear FIRE, must be 1.
NUMBER OF TRIPS	Desired number	Integer	Used with ACCEPT TRANSPORT order.

III-2-A-44

Figure III-2-A-5. DSL Order Modifiers (Continued)

Modifier	Data Type	Data Format	Comments
NUMBER OF VOLLEYS	Volleys	Integer	
ON	Single coordinate pair	x-y, Integer	x and y are integers measured in meters.
OVER	Coordinate pairs	$x_1-y_1, \dots, x_n-y_n$ Integer	$x_i$ and $y_i$ are integers measured in meters. $n \leq 8$ except for area reconnaissance, in which case n must be 4.
PRIORITY	Movement or engineer task priority	1, 2, 3 or 4	
REINFORCING UNIT	UID	Character string	Eight characters beginning with B or R
STOP TASK	Barrier or facility identification code.	Six alphanumeric characters.	First three characters are alpha, last three numeric. One of two orders (not modifiers) requiring data.
TARGET NUMBER	Target index from intelligence report.	Five to seven digits.	
TO	Coordinate pairs	$x_1-y_1, \dots, x_n-y_n$ Integer	$x_i$ and $y_i$ are integers measured in meters. For AIRMOBILE ASSAULT order, $n \leq 4$ . For ADVANCE order $n=1$ . Otherwise $n \leq 8$ .
TYPE	Unit type designator (UTD)	Alphanumeric string	Four characters.
UNIT	Unit identification	Alphanumeric string	Eight characters beginning with B or R.
UNTIL	Absolute time	(see modifier AT TIME)	

III-2-A-45

Figure III-2-A-5. DSL Order Modifiers (Concluded)

NOTES

III-2-A-46



## APPENDIX B

### DIVWAG SCENARIO LANGUAGE COMPILER PROGRAM DESCRIPTIONS

1. INTRODUCTION. This appendix contains detailed descriptions of the routines comprising the DIVWAG scenario language (DSL) compiler and the common area used by those routines.

2. COMMON DSLMN. This common area is used by most of the DSL compiler routines. The variable definitions listed below are consistent throughout the compiler.

<u>Name</u>	<u>Description</u>
NFERR	Total number of fatal errors found in input data.
NWERR	Total number of warning errors found in input data.
DEBUG	Debug print switch.
ISTMT	Pointer to the first character in the array STMT that was processed.
MXSTMT	Maximum number of characters allowed in the STMT array.
STMT	Array containing the statement being processed.
LOCCM	Location of the first comma in the statement.
LOCCLN	Location of the colon in the statement.
LOCPER	Location of the period in the statement.
MXORD	Maximum number of orders allowed in array OARY.
ORDER	Array containing the first four characters of each order.
ORORN	Array containing the order number (NORD) associated with each order.
MXMDFR	Maximum number of modifiers in the MDFR array.
MDFR	Array containing the first four characters of each modifier.
MXNUM	Maximum number of unit scenarios plus battle paragraphs allowed.

<u>Name</u>	<u>Description</u>
UBT	Unit-battle directory table, containing the unit scenario information (loaded from the front) to UNTPT and the battle paragraph information (loaded from the back) to BATPT.
UNTPT	Pointer to the last position in unit battle table used to store unit scenario information.
BATPT	Pointer to the first position in unit battle table used to store battle paragraph information.
UBFLG	Unit scenario-battle paragraph indicator.
CONORD	Conditional-order indicator.
OARY	Order array built by the compiler.
MXOA	Maximum number of orders allowed in each unit scenario or battle paragraph.
JOA	Pointer to the last order placed in the ORDR array.
LABEL	Array containing the labels associated with the orders.
RECPT	Pointer to the last record used in the order file.
ORDFIL	DIVWAG data file identifying number of the order file.
IN	Logical number of input file.
OUT	Logical number of output file.
LNCNT	Total number of lines that have been written on the current page.
MXLN	Maximum number of lines per page.
PERBG	Beginning time of the period in centiminutes.
PERDAY	Day period begins.
PERHR	Hour period begins.
PERMIN	Minute period begins.
PERED	Day period ends.

<u>Name</u>	<u>Description</u>
PEREH	Hour period ends.
PEREM	Minute period ends.
XMAX	Maximum value of X coordinate.
YMAX	Maximum value of Y coordinate.

3. COMMON CONST. This common area is used by many of the DSL compiler routines. It contains the hollerith constants, listed below, that are stored in the first character of the word with the remaining word blank filled.

<u>Name</u>	<u>Description</u>
IPER	Period (.)
ICOM	Comma (,)
ICLN	Colon (:)
IBLNK	Blank ( )
IDASH	Dash or minus sign (-)
INTGR	A 10-word array containing the digits (0-9).
IALPH	A 26-word array containing the alphabet (A-Z).

4. COMMON ONE. This common area is used by routines calling any of the input/output routines.

<u>Name</u>	<u>Description</u>
IFNT	File name table.
IER	Error code.

#### 5. ROUTINE BLOCKD:

a. Purpose. This routine initializes some of the variables for common DSLMN and common CONST.

b. Input Variables. None.

c. Output Variables:

<u>Name</u>	<u>Initial Value</u>	<u>Name</u>	<u>Initial Value</u>	<u>Name</u>	<u>Initial Value</u>
MXSTMT	400	MXNUM	1023	MXOA	100

<u>Name</u>	<u>Initial Value</u>	<u>Name</u>	<u>Initial Value</u>	<u>Name</u>	<u>Initial Value</u>
ORDFIL	55	IN	60	OUT	61
MXLN	55	MXORD	25	MXMDFR	32
ORDR(1)	4HFLYB	ORDR(2)	4HSTAY	ORDR(3)	4HGOTO
ORDR(4)	4HMOVE	ORDR(5)	4HFIRE	ORDR(6)	4HPREP
ORDR(7)	4HADVA	ORDR(8)	4HENG	ORDR(9)	4HRECO
ORDR(10)	4HACCE	ORDR(11)	4HAIRM	ORDR(12)	4HASSI
ORDR(13)	4HASSU	ORDR(14)	4HBREA	ORDR(15)	4HBUIL
ORDR(16)	4HDETA	ORDR(17)	4HJOIN	ORDR(18)	4HLOIT
ORDR(19)	4HMISS	ORDR(20)	4HRELE	ORDR(21)	4HREMO
ORDR(22)	4HRETA	ORDR(23)	4HSTOP	ORDR(24)	4HTERM
ORDR(25)	4HWITH	ORDN(1)	7	ORDN(2)	1
ORDN(3)	-1	ORDN(4)	3	ORDN(5)	9
ORDN(6)	2	ORDN(7)	6	ORDN(8)	15
ORDN(9)	8	ORDN(10)	20	ORDN(11)	41
ORDN(12)	23	ORDN(13)	24	ORDN(14)	43
ORDN(15)	42	ORDN(16)	26	ORDN(17)	25
ORDN(18)	32	ORDN(19)	39	ORDN(20)	31
ORDN(21)	44	ORDN(22)	40	ORDN(23)	45
ORDN(24)	37	ORDN(25)	5	MDFR(1)	4HAIRC
MDFR(2)	4HATAL	MDFR(3)	4HATSP	MDFR(4)	4HATTI
MDFR(5)	4HATWI	MDFR(6)	4HBARR	MDFR(7)	4HBEGI
MDFR(8)	4HBRID	MDFR(9)	4BYbb	MDFR(10)	4HCOMP
MDFR(11)	4H-DEP	MDFR(12)	4HDESI	MDFR(13)	4HDIRE
MDFR(14)	4HFACI	MDFR(15)	4HFORb	MDFR(16)	4HGENE
MDFR(17)	4HHEIG	MDFR(18)	4HIMPA	MDFR(19)	4HINBA

<u>Name</u>	<u>Initial Value</u>	<u>Name</u>	<u>Initial Value</u>	<u>Name</u>	<u>Initial Value</u>
MDFR(20)	4HMAND	MDFR(21)	4HMIXb	MDFR(22)	4HMUNI
MDFR(23)	4HNUMB	MDFR(24)	4HONbb	MDFR(25)	4HOVER
MDFR(26)	4HPRIO	MDFR(27)	4HREIN	MDFR(28)	4HTARG
MDFR(29)	4HTObb	MDFR(30)	4HTYPE	MDFR(31)	4HUNIT
MDFR(32)	4HUNTI	IPER	1H.	ICOM	1H,
ICLN	1H:	IBLNK	1H	IDASH	1H-
INTGR(1)	1H1	INTGR(2)	1H2	INTGR(3)	1H3
INTGR(4)	1H4	INTGR(5)	1H5	INTGR(6)	1H6
INTGR(7)	1H7	INTGR(8)	1H8	INTGR(9)	1H9
INTGR(10)	1H0	IALPH(1)	1HA	IALPH(2)	1HB
IALPH(3)	1HC	IALPH(4)	1HD	IALPH(5)	1HE
IALPH(6)	1HF	IALPH(7)	1HG	IALPH(8)	1HH
IALPH(9)	1HI	IALPH(10)	1HJ	IALPH(11)	1HK
IALPH(12)	1HL	IALPH(13)	1HM	IALPH(14)	1HN
IALPH(15)	1HO	IALPH(16)	1HP	IALPH(17)	1HQ
IALPH(18)	1HR	IALPH(19)	1HS	IALPH(20)	1HT
IALPH(21)	1HU	IALPH(22)	1HV	IALPH(23)	1HW
IALPH(24)	1HX	IALPH(25)	1HY	IALPH(26)	1HZ

#### 6. ROUTINE DSLCMP:

a. Purpose. This is the controlling routine of the DSL compiler. It makes the major decisions to store a unit scenario or battle paragraph and initialize the arrays so another may be built, to initiate the second pass of the compiler, to dump the data file if requested, and to abort the run if fatal errors are discovered.

b. Input Variables. DSL common, NFERR, DBUG, ISTMT, STMT, LOCCLN, UBT, UNTPT, BATPT, UBFLG, OARY, JOA, LABEL and OUT.

c. Output Variables. DSL common variables, NFERR, NWERR, ISTMT, UBT, UNTPT, BATPT, UBFLG, OARY, and LABEL.

d. Logical Flow (Figure III-2-B-1):

(1) Block 1. DSLINT is called to perform initialization of necessary variables.

(2) Block L2000. This block initializes the label array (LABEL), order array (OARY), JOA and UBFLG each time a unit scenario or battle paragraph is to be built.

(3) Block L3000. If the next statement is to be read from cards, control is transferred to block L3005; otherwise, control passes to block L3010.

(4) Block L3005. RDSTMT is called to read the next statement from cards.

(5) Block 2. If this statement is FINIS, control is passed to block 4.

(6) Block 3. If the statement begins with ID or BATTLE, it indicates the beginning of a new unit scenario or battle paragraph and control is transferred to block L2000.

(7) Block L3010. The type of statement is further determined and the appropriate action taken to process it. If it is a comment, no further action is necessary. If it is the end of a unit scenario or battle paragraph STOW is called to store it on the DSL data file. If it is an order, UBFLG is interrogated; UNTORD is called if it is one or BATORD is called if it is two.

(8) Block 4. PASS2 is called to complete the compilation.

(9) Blocks 5 and 6. If DEBUG was specified on the DSL card, DBUG will be true and DUMPF will be called to dump the DSL data file.

(10) Block 7. The total number of fatal errors and warning errors detected by the compiler are written.

(11) Blocks L8500 and 8. If fatal errors were detected by the compiler, the run will be aborted by generating a system error mode 1, the DSL data file will be invalidated, and an error message is printed.

7. ROUTINE DSLINT:

a. Purpose. This routine initializes the areas of common that are not initialized by the DSL block data routine. It reads the DSL, start of period, and start of game cards and initializes the variables obtained from them. It also loads the unit identification and unit location tables.

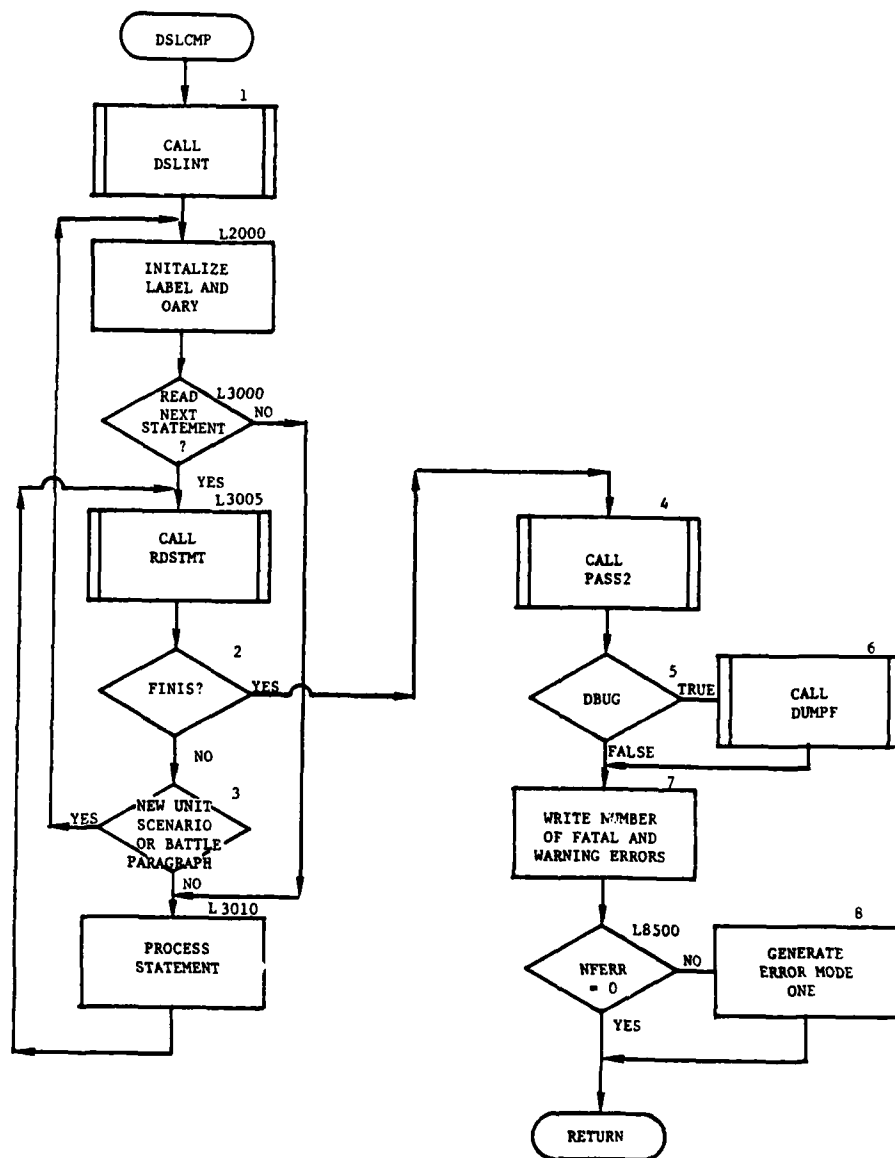


Figure III-2-B-1. Routine DSLCMP

<u>Name</u>	<u>Source</u>	<u>Contents</u>
PERDAY	Card	Beginning period day.
PERHR	Card	Beginning period hour.
PERMIN	Card	Beginning period minute.
IREC1(6)	Card	Start of period/start of game flag.
IREC1(4)	Card	Period length.
DEBUG	Card	DSL debug option.

c. Output Variables:

(1) DSL Common Block Variables. IFNT, IER, LNENT, PGNO, NFERR, NWERR, DBUG, PERED, PEREH, PEREM, PERDAY, PERHR, PERMIN, XMAX, YMAX, RECPT, UNTPT, BATPT, and UBT.

(2) Other Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
IREC1(4)	Call	Period length.
IREC1(6)	Call	Start of period/start of game flag.
UIDTAB	TWO	Unit identification table.
UNTLOC	TWO	Unit location table.
BPOINT	TWO	Pointer to last Blue unit in UIDTAB.
RPOINT	TWO	Pointer to first Red unit in UIDTAB.

d. Logical Flow (Figure III-2-B-2):

(1) Blocks 1 and 2. The array IFNT is checked to determine if the order file has been created at a proper size. If it has not been created, it is created.

(2) Blocks L1002 and 3. The line count (LNCNT), page number (PGNO), number of fatal errors (NFERR), and number of warning errors (NWERR), are initialized and DSL COMPILATION and PAGE1 are written on the first line of a new page.

(3) Block 4. The first three cards of the DSL deck are read and the values of the beginning day, hour, and minute (PERDAY, PERHR, PERMIN), the period ending day, hour, and minute (PERED, PEREH, PEREM), and the period length (IREC1(4)) are initialized. If the DEBUG option is found on the first card, DEBUG is set to TRUE. It also sets the start of game/start of period flag (IREC1(6)) to three or one respectively.



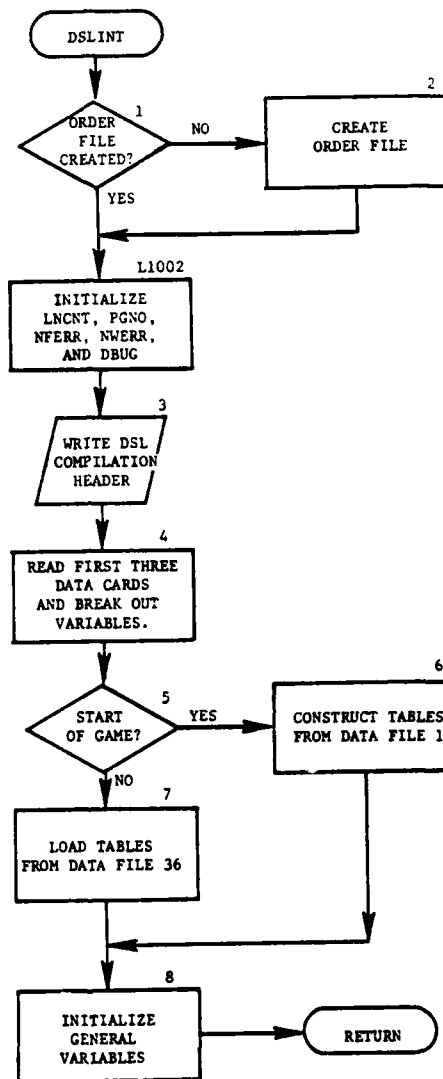


Figure III-2-B-2. Routine DSLINT

(4) Blocks 5, 6, and 7. IRECl(6) is checked to determine if it is equal to three. If it is, the UIDTAB and UNTLOC tables are constructed; otherwise, these tables are loaded from data file 36.

(5) Block 8. The maximum values of the X and Y coordinates (XMAX and YMAX), the pointer to the last record used in the order file (RECPT), the pointer to the last unit loaded (UNTPT), and the pointer to the last battle loaded (BATPT) are initialized. Also, the appropriate words of the unit and battle directory table (UBT) are set to blanks and zeros.

#### 8. ROUTINE RDSTMT:

a. Purpose. This routine reads and prints DSL data cards. A card is read until a period is encountered or the STMT array is filled. If the STMT array is filled, a period is placed in it's last position and the routine reads until a period or colon is encountered. The routine also returns the position of the period, the position of a colon if one is found, and the position of the first comma if one is found.

##### b. Input Variables:

(1) DSL Common Block Variables.

(2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ISAV	Call	Error indicator: 1 = period was left out in previous DSL statement and colon was found in next DSL statement, ICARD contains the previous card image. A new card is not read. 0 = normal entry. Read new DSL statement.
ICARD	Card	DSL order statement.

##### c. Output Variables:

(1) DSL Common Block Variables. LOCCM, LOCCLN, LOCPER, ISTMT, NFERR, INWERR, STMT, and LNCNT.

(2) Other Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
ICARD	Print	DSL order statement.

##### d. Logical Flow (Figure III-2-B-3):

(1) Block L1000. Set the period, comma, color, and STMT location pointers and the number of characters counter are set to zero.

(2) Blocks 1 and L1010. If ISAV is less than or equal to zero, a new card is read.

AD-A112 688

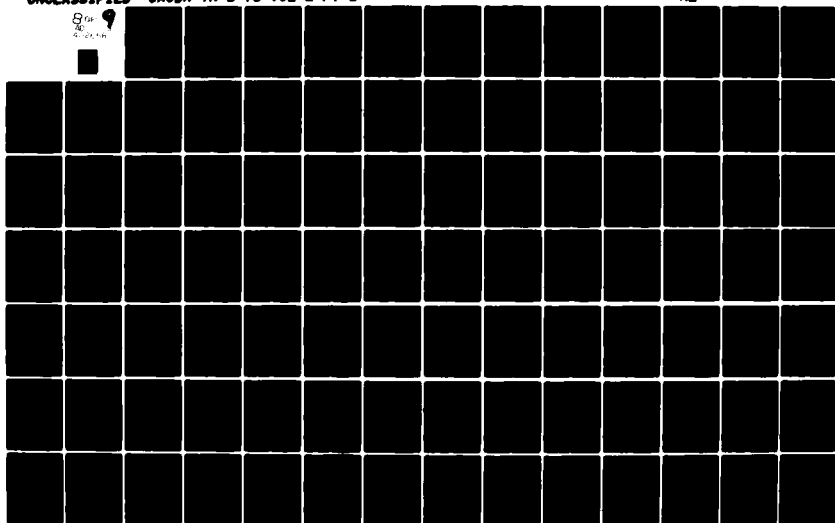
ARMY COMBINED ARMS COMBAT DEVELOPMENTS ACTIVITY FORT--ETC F/8 18/7  
DIVVAG MODEL DOCUMENTATION. VOLUME II. PROGRAMMER/ANALYST MANUAL--ETC(U)  
JUL 76

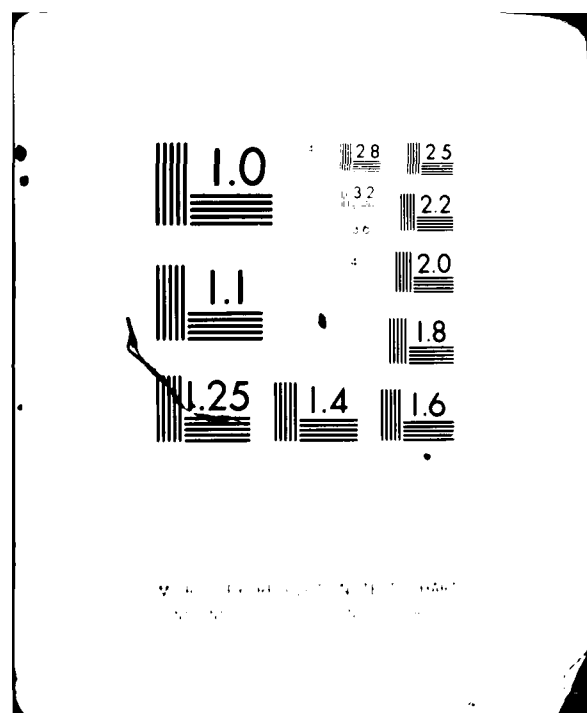
UNCLASSIFIED

CACDA-TR-8-76-VOL-2-PT-1

NL

8-10  
9  
4-12-10





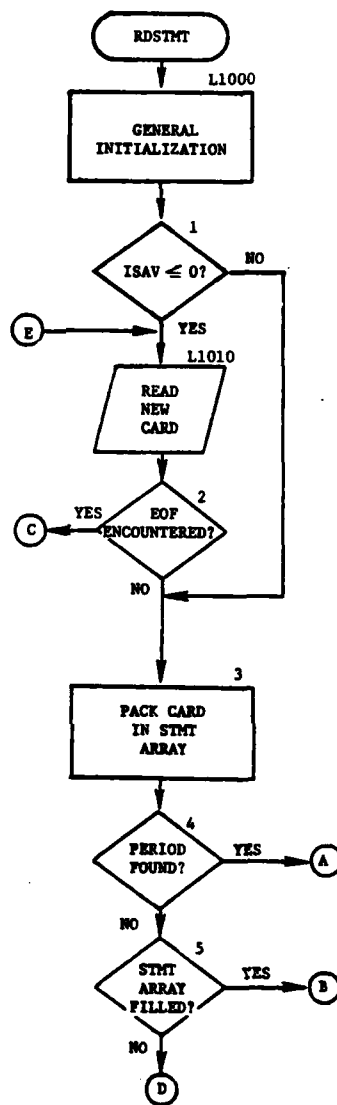


Figure III-2-B-3. Routine RDSTMT (Continued on Next Page)

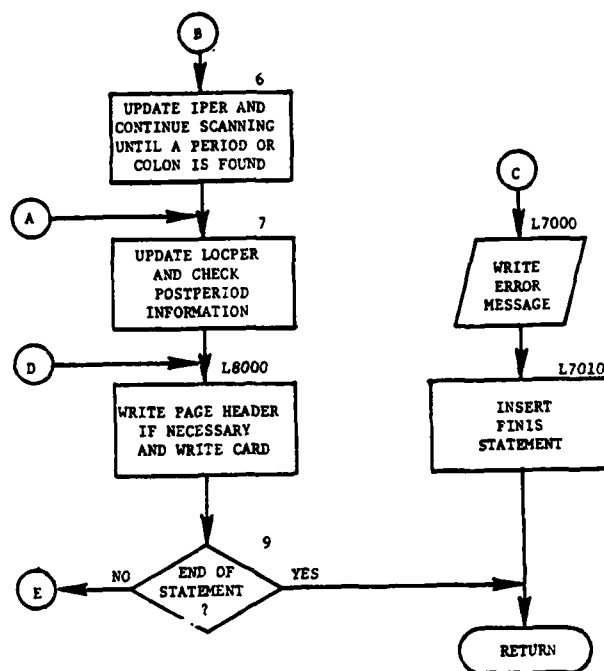


Figure III-2-B-3. Routine RDSTMT (Concluded)

(3) Block 2. Control transfers to block L7000 if an end of file is encountered.

(4) Block 3. As the contents of CARD are packed into START, each character is checked to determine if it is a comma, period or colon. If a colon is found, its location (in STMT) is stored in LOCCLN. If a comma is found, LOCCM is compared to zero to determine if it is the first comma in the statement; if it is first, its location is stored in LOCCM.

(5) Blocks 4 and 5. If a period is found, control transfers to block 7; otherwise, it is determined if the STMT array is full and control transfers to block 6 if it is.

(6) Block 6. A period is put in the last location of the STMT array, and reading is continued until a period or colon is found. If a colon is found, ICARD is saved and used as the first card the next time RDSTMT is called.

(7) Block 7. The period location pointer (LOCPER) is updated and checked for nonblank characters following the period. A warning message is printed if nonblank characters are found.

(8) Blocks L7000 and L7010. If an end of file was encountered while reading the DSL data cards, an error message is written and the six characters FINIS. are inserted in STMT.

(9) Block L8000. A new page header is written if the statement belongs to a different force or to a battle paragraph rather than a force.

(10) Block 9. If an end of statement has been detected, control is returned to the calling routine; otherwise, control is transferred to block L1010.

#### 9. ROUTINE BATORD:

a. Purpose. This routine processes the battle paragraph statements by storing the participating unit list and the corresponding labels and calling COND to process the conditionals.

b. Input Variables. DSL common block variables LOCCLN, ISTMT, BATPT, LOCPER, NFERR, IBLNK, STMT, ICOM, and JOA.

c. Output Variables. DSL common block variables OARY, NFERR, JOA and ISTMT

d. Logical Flow (Figure III-2-B-4):

(1) Block 1. This block decides whether the battle order is a conditional or unit list.

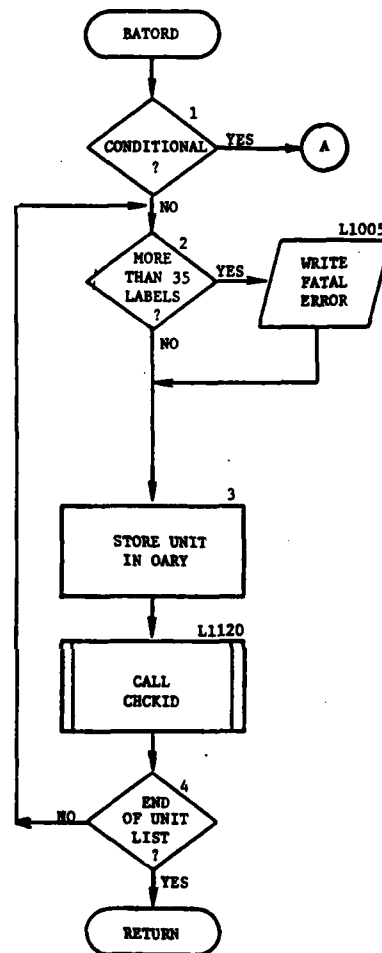


Figure III-2-B-4. Routine BATORD (Continued on Next Page)



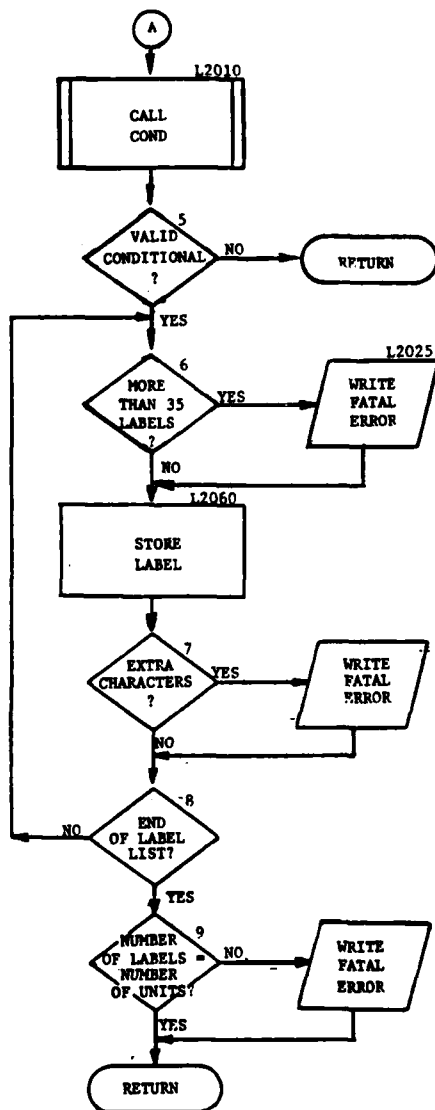


Figure III-2-B-4. Routine BATORD (Concluded)

(2) Block 2 and L1005. If there are more than 35 units in this battle, the number of fatal errors (NFERR) is incremented by one and an appropriate error message is written.

(3) Block 3. The unit list is stored in records one to four of OARY, 10 unit identifications per record.

(4) Block L1120. CHCKID is called to find errors in the unit identifications.

(5) Block 4. If the list of units is completed, control is returned to the calling routine; otherwise, another unit identification is processed.

(6) Blocks L2010 and 5. COND is called to return the value IAO. If IAO is less than zero, control is returned to the calling routine.

(7) Blocks 6 and L2025. If the number of labels is greater than 35, NFERR is incremented by one and an appropriate error message is written.

(8) Block L2060. The labels are stored OARY.

(9) Blocks 7 and L2115. If a label is followed by extra characters, NFERR is incremented by one and an appropriate error message is written.

(10) Block 8. If the label list is not completed, control returns to block 6.

(11) Blocks 9 and 10. If the number of labels equals the number of units, control is returned to the calling routine. If they are not equal, NFERR is incremented by one and a fatal error message is written.

#### 10. ROUTINE UNTORD:

a. Purpose. This routine processes unit scenario orders and conditionals. It determines the type of order and calls the appropriate order routine, stores the statement label if there is one, and calls COND if necessary.

b. Input Variables. DSL common block variables LOCCLN, NFERR, STMT, LABEL, IALPH, OARY, IBLNK, ISTMT, ORDR, MXORD, JOA, and LOCPER.

c. Output Variables. DSL common block variables LABEL, NFERR, JOA, and OARY.

#### d. Logical Flow (Figure III-2-B-5):

(1) Block 1. If a colon is present in the statement, control transfers to block 2, or to block L2010 otherwise.

(2) Blocks 2 and 3. If the label is legal (less than four characters), processing continues. If it is not legal, an error message is written.

(3) Block L1020. The label is stored in the array LABEL.

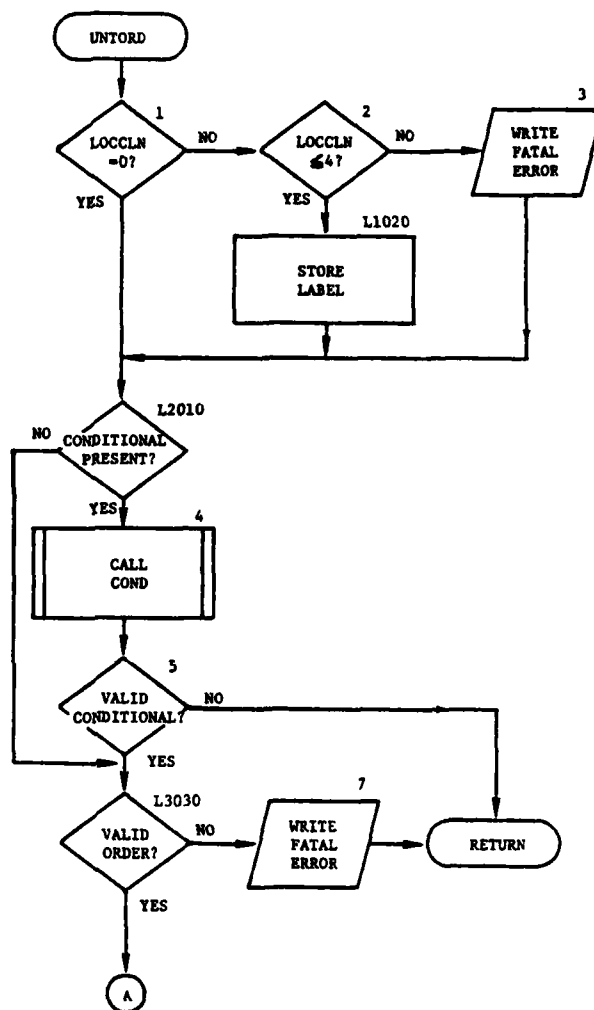


Figure III-2-B-5. Routine UNTORD (Continued on Next Page)

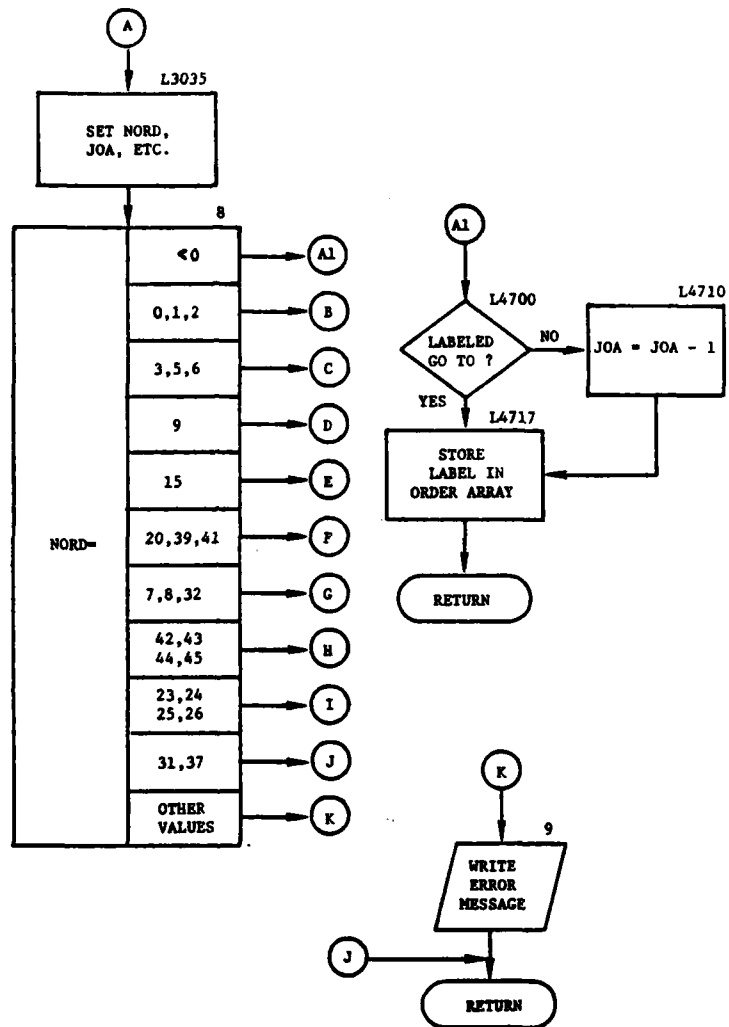


Figure III-2-B-5. Routine UNTORD (Continued)

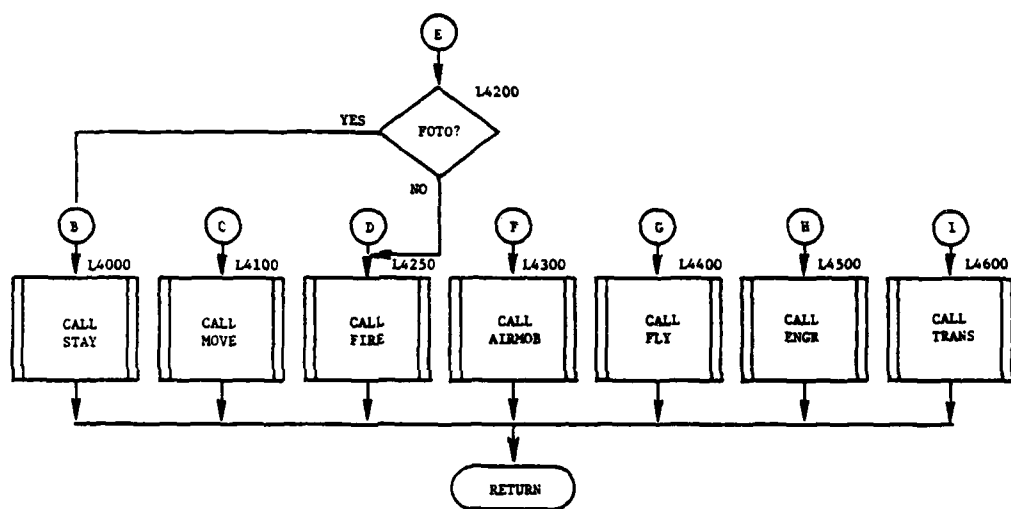


Figure III-2-B-5. Routine UNTORD (Concluded)

(4) Blocks L2010 and 4. If the statement contains a conditional, COND is called.

(5) Block 5. COND returns parameter IAO. If this parameter is less than zero, the conditional is not valid, and processing of the statement is aborted by returning control to the calling routine.

(6) Blocks L3030 and 7. If the order cannot be matched in array ORDR, an error message is written stating the order is invalid. If there is a match, the order type is determined by the value stored in the same position in ORDRN.

(7) Block L3035. NORD is set equal to the identifying number associated with this particular order.

(8) Block 8. NORD determines which order routine to call.

(9) Block 9. An error message is written if the value of NORD is invalid.

(10) Blocks L4700, L4710, and L4717. These blocks are for GOTO orders. If the order is labeled or contains a conditional, the GOTO order requires a separate statement. Otherwise, GOTO is put at the end of the previous order.

(11) Block L4200. This block checks for a fire on targets of opportunity (FOTO) order. If it is a FOTO order, routine STAY is called.

(12) Blocks L4000, L4100, L4250, L4300, L4400, L4500, and L4600. These blocks call the appropriate order routines.

#### 11. ROUTINE COND:

a. Purpose. This routine decodes the conditional clauses of the unit scenario and battle paragraph orders. Each clause of a conditional is considered separately.

b. Input Variables. DSL common variables NFERR, NWERR, ISTMT, STMT, LOCCM, JOA, OUT and OARY.

c. Output Variables:

(1) DSL Common Variables. NFERR, NWERR, ISTMT, OARY, and JOA.

(2) Other Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
IOA	Call	Conditional clause indicator: Less than zero = fatal error in processing conditional 1 = clauses are connected by AND 2 = clauses are connected by OR.

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
KNTOA	Call	Number of clauses in statement.

d. Logical Flow (Figure III-2-B-6):

(1) Block L1000. Initialization in this block checks the conditional to be sure a comma is in the statement and writes an error message and aborts processing if it is missing. KNTOA and IOA are initialized and it is determined if each clause begins with if or when.

(2) Block L1021. If the clause separators--and/or--are present in the conditional, the locations are stored. A check is made to assure both "and and "or" do not appear in the same conditional.

(3) Blocks 1, 2, 3, and L1030. If the first character of the conditional is not a B or R, control is transferred to block L3000. If an R is found, another check must be made to determine if the relative humidity condition is being checked. If this is the case, control is transferred to block L2000; otherwise, the unit identification is stored and control passes to block L1045.

(4) Blocks L1045 and 4. The conditional is checked to determine if 'not' is present. If it is, the indicator is stored in OARY.

(5) Blocks L1200 and 5. The conditional is checked for 'HALTED AT' or 'AT LOCATION'. If one is found, the coordinate pair associated with it is stored in OARY and control passes to block L7000.

(6) Blocks L1300 and 6. If the unit's activities of movement, stopping, or firing or of being assessed are to be determined the indicator is stored and control is transferred to block L7000.

(7) Block L1500. If an equipment type conditional is found, the equipment type is stored and control is transferred to block L7000.

(8) Blocks L1600 and 8. If the condition of class III, class V or present strength is detected, the indicator is stored and control is passed to block L2000.

(9) Block L1700. If weather conditionals--cloud cover, fog index, relative humidity, temperature, visibility, wind speed or direction, and precipitation--are not found, control is transferred to block L1800; otherwise, control passes to block L2000.

(10) Block L1800. This clause could not be recognized as a conditional; so, an error message is written and control is transferred to block L7000.

(11) Blocks L2000 and 9. If 'not' is found in the conditional, its indicator is stored in OARY.

(12) Blocks L2200 and 10. If a 'greater than', 'equal to' or 'less than' appears in the conditional, the appropriate indicator is stored.

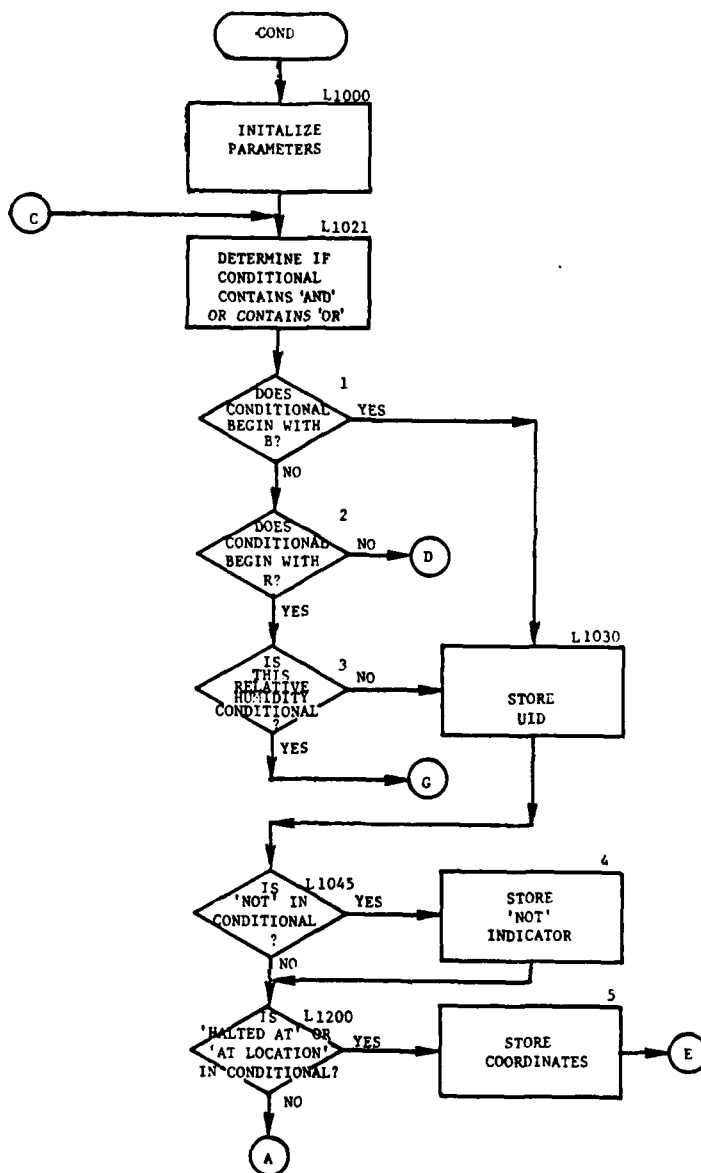


Figure III-2-B-6. Routine COND (Continued on Next Page)



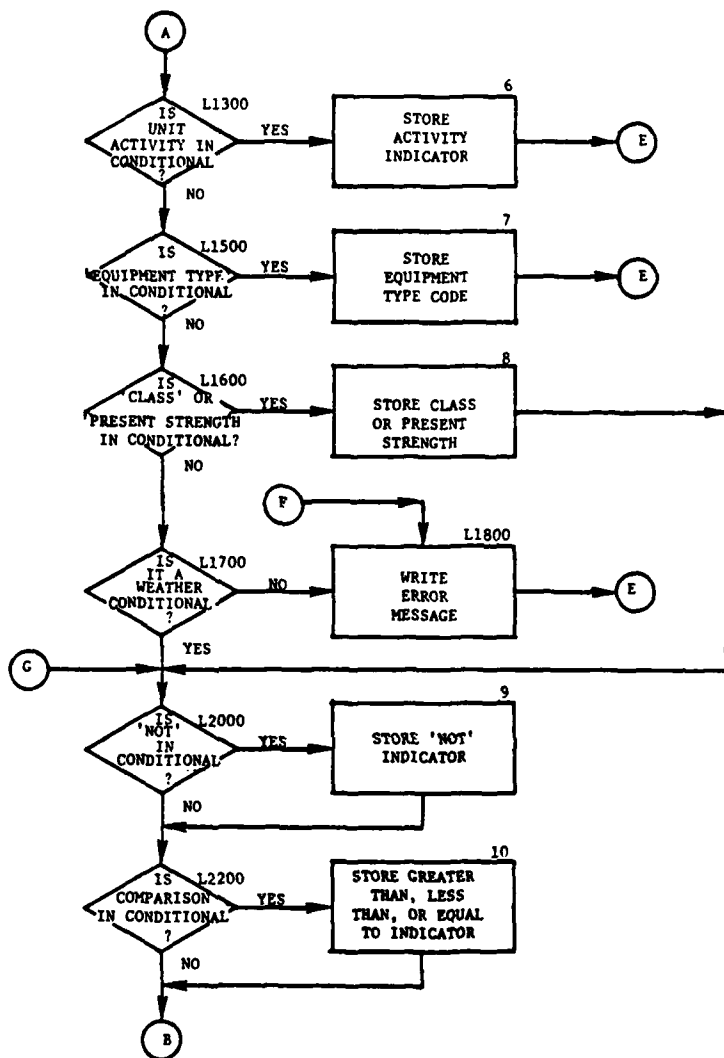


Figure III-2-B-6. Routine COND (Continued)

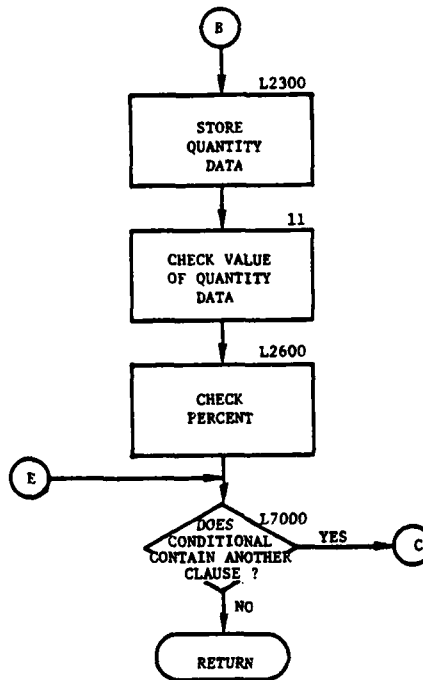


Figure III-2-B-6. Routine COND (Continued)

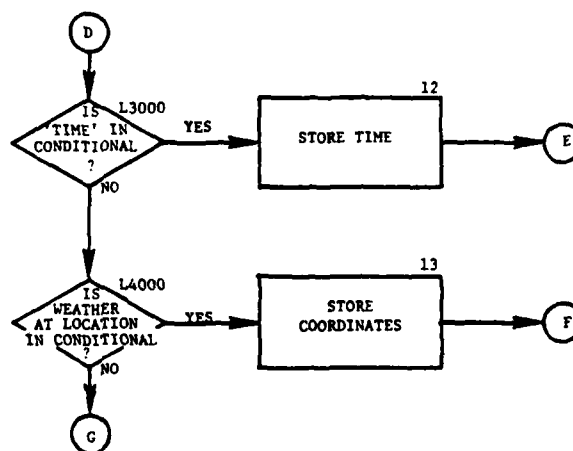


Figure III-2-B-6. Routine COND (Concluded)

(13) Block L2300. The numeric quantity data associated with the conditional is decoded and stored.

(14) Block 11. The value of the quantity data must fall in prescribed ranges. The allowed values are:

- . visibility index, 1-9 inclusive
- . cloud cover and relative humidity, 1-100
- . precipitation index, 0-2
- . temperature gradient, 1-4
- . wind direction, 0-360
- . fog index 0-1
- . wind speed, positive.

If values are found that are not within these ranges an error message is written.

(15) Block L2600. Equipment type and present strength are compared to an absolute value or to a percentage of the amount authorized to the unit. If percent is found, it is stored.

(16) Block L7000. If the conditional contains another clause, control transfers to block L1021; otherwise, it returns to the calling routine.

(17) Blocks L3000 and 12. If the conditional contains a check against time, the time is decoded and stored, and control is transferred to block L7000.

(18) Blocks L4000 and 13. If the condition to be considered is weather at a particular location, the coordinates are stored, and control is passed to block L2000; otherwise, control goes to block L1800.

## 12. ROUTINE AIRMOB:

a. Purpose. This routine builds the list of modifiers for the orders accept transport, airmobile assault, mission is, and retain. If a required or exclusive modifier is missing in an order, an appropriate error message is written.

b. Input Variables. DSL common block variables OARY, NFERR, MDFR, and JOA.

c. Output Variables. DSL common block variable NFERR.

d. Logical Flow (Figure III-2-B-7):

(1) Block 1. Determine the order type (accept transport, airmobile assault, mission is, retain). Store the modifier indexes that correspond to the particular order in the array MDF.

(2) Block L1000. KRAKMD returns MDF with the value zero replacing the index of each modifier that was found in the order.

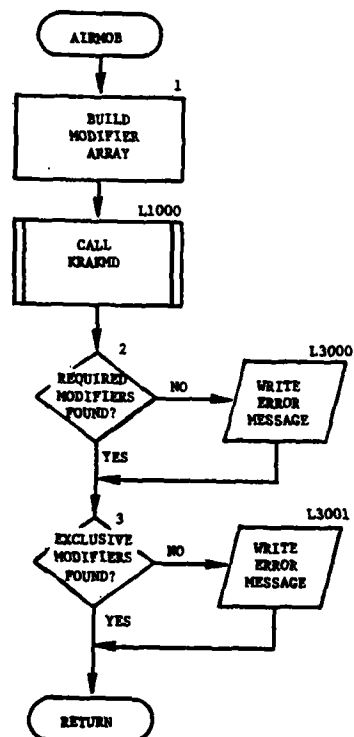


Figure III-2-B-7. Routine AIRMOB

(3) Block 2. MDF is checked to verify that all required modifiers were found in the order. If not, block L3000 is executed.

(4) Block 3. MDF is checked to verify that one of each exclusive modifier set was found in the order. Block L3001 is executed for each set that is missing.

(5) Block L3000. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written listing each required modifier omitted in the order.

(6) Block L3001. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written listing each exclusive modifier set omitted in the order.

### 13. ROUTINE ENGR:

a. Purpose. This routine builds the list of modifiers for the orders build, breach, and remove. If any exclusive modifiers are missing or there are too many modifiers input, an error message is printed.

b. Input Variables. DSL common block variables OARY, NFERR, MDFR, and JOA.

c. Output Variables. DSL common block variable NFERR.

d. Logical Flow (Figure III-2-B-8):

(1) Block 1. Determine the order type (build, breach, remove). Store the modifier indexes that are used with the particular order in the array MDF.

(2) Block L302. KRAKMD returns MDF with the value zero replacing the index of each modifier that was found in the order.

(3) Block 2. MDF is checked to verify that only one of each exclusive modifier set is in the order. If there are too many exclusive modifiers in the order, block L2000 is executed.

(4) Block 3. MDF is checked to verify that one of each exclusive modifier set was found. Block L2001 is executed for each set that is missing.

(5) Block L2000. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written.

(6) Block L2001. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written listing each exclusive modifier set not found.

### 14. ROUTINE FIRE:

a. Purpose. This routine builds the list of modifiers for the orders fire (nuclear), fire (conventional), and engage. If any required or exclusive modifiers are missing, an error message is printed.

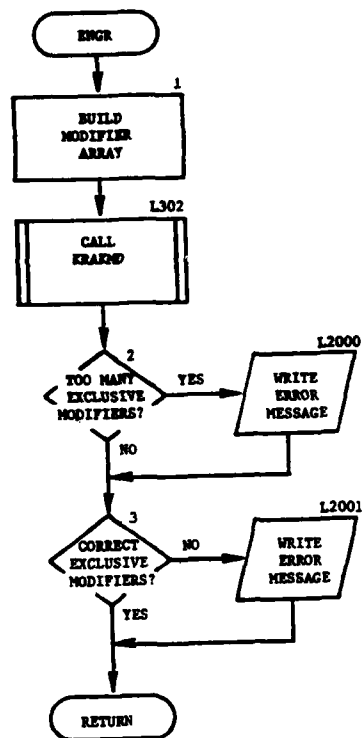


Figure III-2-B-8. Routine ENGR

b. Input Variables. DSL common block variables OARY, NFERR, MDFR, and JOA.

c. Output Variables. DSL common block variable NFERR.

d. Logical Flow (Figure III-2-B-9):

(1) Block 1. Determine the order type (fire, nuclear; fire, conventional; engage). Store the modifier indexes that correspond to the particular order in the array MDF.

(2) Block L1000. KRAKMD returns MDF with the value zero replacing the index of each modifier that was found in the order.

(3) Block 2. MDF is checked to verify that all required modifiers were found. Block L3000 is executed for each one that is missing.

(4) Block 3. MDF is checked to verify that exclusive modifiers of each set was found. Block L3010 is executed for each set that is missing.

(5) Block L3000. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written listing each required modifier not found.

(6) Block L3010. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written listing each exclusive modifier set not found.

#### 15. ROUTINE FLY:

a. Purpose. The routine checks all of the required modifiers for the orders, fly, loiter, and reconnoiter. If any required modifiers are missing, an error message is printed for each required modifier not found in the order.

b. Input Variables. DSL common block variables OARY, NFERR, MDFR, and JOA.

c. Output Variables. DSL common block variable NFERR.

d. Logical Flow (Figure III-2-B-10):

(1) Block 1. Determine the order type (fly, loiter, or reconnaissance). Store the indexes of the modifiers that apply to the particular order in the array MDF.

(2) Block L1001. KRAKMD zeros the positions in MDF corresponding to each modifier found in the order.

(3) Block 2. MDF is checked to verify that all of the required modifiers were found. Block L2000 is executed for each modifier that is missing.



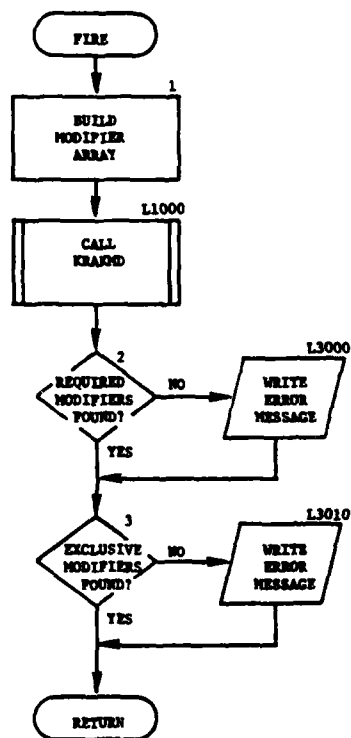


Figure III-2-B-9. Routine FIRE

III-2-B-31

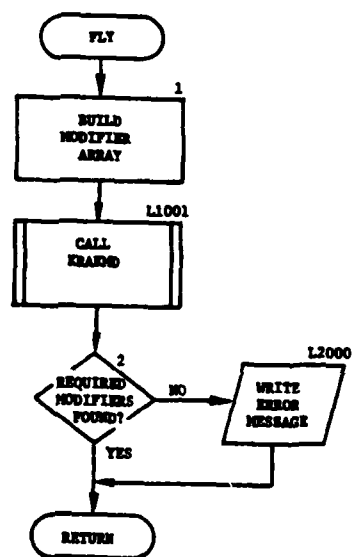


Figure III-2-B-10. Routine FLY

(4) Block L2000. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written listing the required modifiers not found.

16. ROUTINE MOVE:

a. Purpose. This routine builds the list of modifiers for the orders move, advance, and withdraw. If a required modifier is missing an error message is written. Also, if the order is advance or withdraw, the routine checks to determine if both width and depth modifiers were found if these optional modifiers were used.

b. Input Variables. DSL common block variables OARY, NFERR, NWERR, MDFR, and JOA.

c. Output Variables. DSL common block variables NFERR and NWERR.

d. Logical Flow (Figure III-2-B-11):

(1) Block 1. Determine the order type (move, advance, withdraw). Store the modifier indexes that correspond to the particular order in the array MDF.

(2) Block 2. KRAKMD returns MDF with the value zero replacing the index of each modifier that was found in the order.

(3) Block 3. MDF is checked to verify that all required modifiers were found in the order. If some were not found or incorrect modifiers were found, block 4 is executed.

(4) Block 4. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written.

(5) Block 5. MDF is checked to verify that both parts of the width-depth pair were found. If both were not found, block L2700 is executed.

(6) Block L2700. The number of warning errors (NWERR) is incremented by one, and a warning error message is written.

17. ROUTINE STAY:

a. Purpose. This routine builds the list of modifiers for the orders stay, prepare, and fire on targets of opportunity (FOTO). If any exclusive modifiers are missing or if more than one is input, an error message is written. Also, the width and depth are checked to determine if both were input if the width-depth optional modifiers are used.

b. Input Variables. DSL common block variables OARY, NFERR, NWERR, MDFR, and JOA.

c. Output Variables. DSL common block variables NFERR and NWERR.

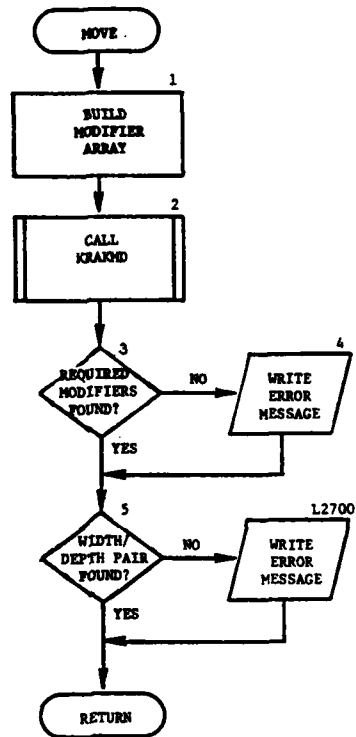


Figure III-2-B-11. Routine MOVE

d. Logical Flow (Figure III-2-B-12):

(1) Block 1. Determine the order type (stay, prepare, fire on targets of opportunity). Store the modifier indexes that correspond to the particular order in the array MDF.

(2) KRAKMD returns MDF with the value zero replacing the index of each modifier that was found in the order.

(3) Block 3. MDF is checked to determine if exclusive modifiers were found in the order. Block 4 is executed if none were found.

(4) Block 4. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written.

(5) Block 5. MDF is checked to determine if more than one exclusive modifier was found in the order. Block L2600 is executed if more than one was found.

(6) Block L2600. The number of warning errors (NWERR) is incremented by one, and a warning error message is written.

(7) Block 6. MDF is checked to see that both parts of the width-depth pair were found in the order. If both were not found, block L2700 is executed.

(8) Block L2700. The number of warning errors (NWERR) is incremented by one, and a warning error message is written.

18. ROUTINE TRANS:

a. Purpose. This routine builds the list of modifiers for the orders assignment is, assume control of, detach, and join. If a required or exclusive modifier is missing, an appropriate error message is written.

b. Input Variables. DSL common block variables OARY, NFERR, MDFR, and JOA.

c. Output Variables. DSL common block variable NFERR.

d. Logical Flow (Figure III-2-B-13):

(1) Block 1. Determine the order type (assignment is, assume control of, detach, or join). Store the modifier indexes that correspond to the particular order in the array MDF.

(2) Block L1000. KRAKMD returns MDF with the value zero replacing the index of each modifier in MDF that was found in the order.

(3) Block 32. MDF is checked to verify that all required modifiers were found in the order. Block L3000 is executed for each one that is missing.

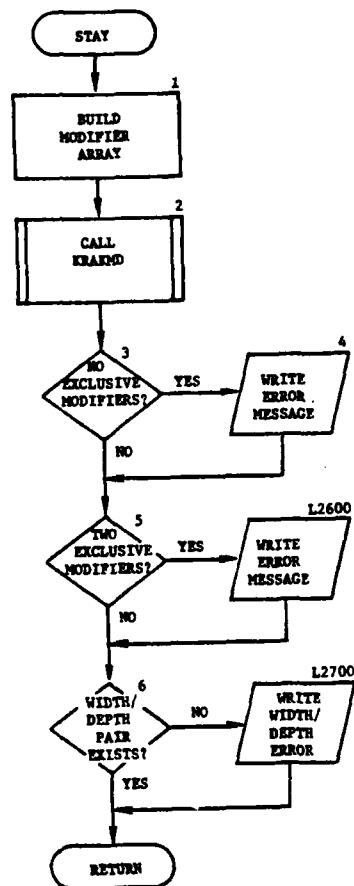


Figure III-2-B-12. Routine STAY

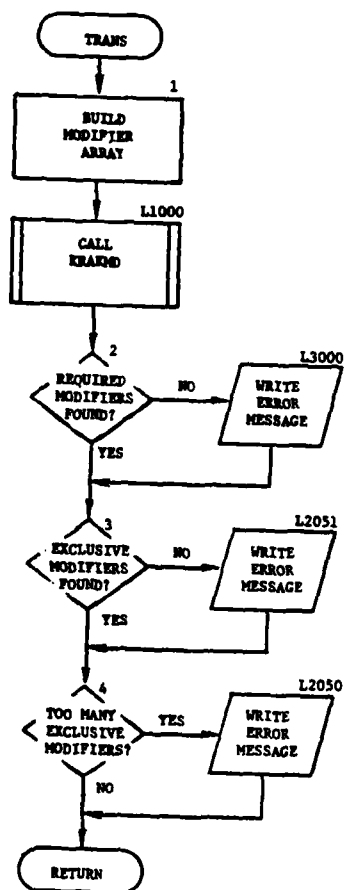


Figure III-2-B-13. Routine TRANS

III-2-B-37

(4) Block 3. MDF is checked to verify that one of each exclusive modifier set was found. Block L2051 is executed for each set that is missing.

(5) Block 4. MDF is checked to see that too many exclusive modifiers were not found. If too many were found, block L2050 is executed.

(6) Block L3000. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written listing each required modifier omitted in the order.

(7) Block L2051. The number of fatal errors (NFERR) is incremented by one, and an appropriate fatal error message is written.

(8) Block L2050. The number of fatal errors (NFERR) is incremented by one, and an appropriate error message is written.

19. ROUTINE KRAKMD:

a. Purpose. This routine decodes the modifiers associated with a particular DSL order. As each modifier is found, an appropriate entry is made in OARY. Error checking is also done where possible.

b. Input Variables:

(1) DSL Common Variables. NFERR, NWERR, ISTMT, STMT, LOOPER, OARY, JOA, OUT, MXMDFX, and PERBG.

(2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Description</u>
MDF	Call	Array containing pointers to the modifiers that may be applied to this order.
NM	Call	Number of modifiers in MDF.

c. Output Variables. DSL common variables NFERR, NWERR, ISTMT, JOA, and OARY.

d. Logical Flow (Figure III-2-B-14):

(1) Block L5010. The number of modifiers (NM) is compared to the maximum number of modifiers (MXMDFR) to assure there is room in the MDF array. If there isn't an error message is printed and NM is set equal to MXMDFR. The first four characters of the modifiers are extracted from the MDFR array and stored one-character-per-word in the MDFX array.

(2) Block L5100. The STMT array is scanned from ISTMT to LOOPER for a numeric character. If one is found, its location is stored in MXC and the numeric string is converted to an integer and stored in IVAL.



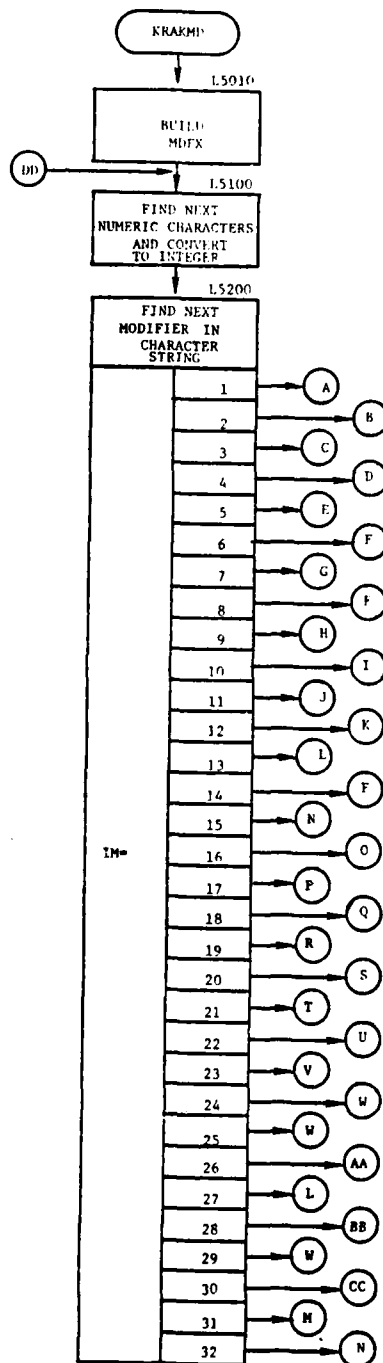


Figure III-2-B-14. Routine KRAKMD (Continued on Next Page)

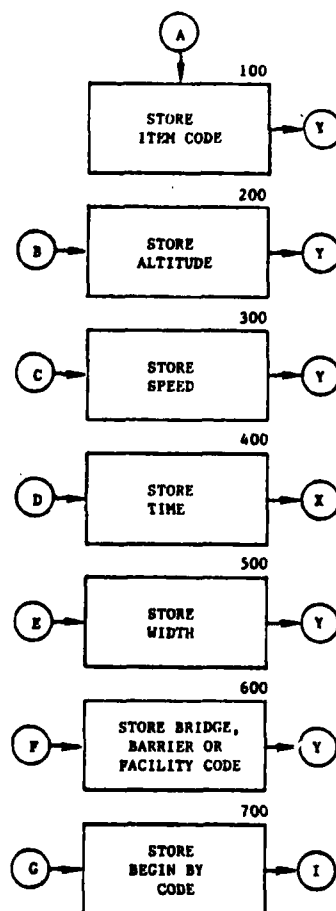


Figure III-2-B-14. Routine KRAKMD (Continued)

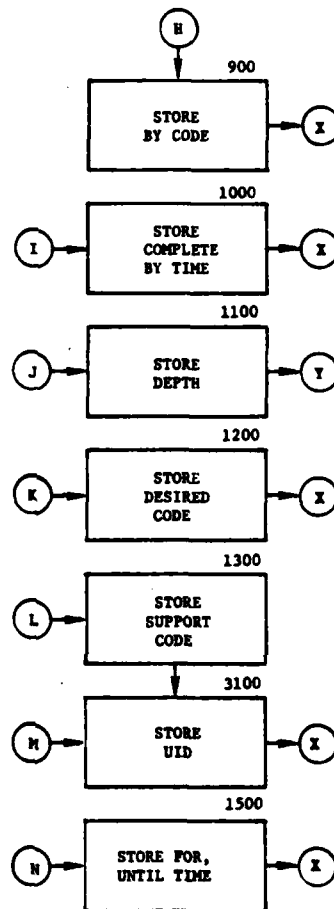


Figure III-2-B-14. Routine KRAKMD (Continued)

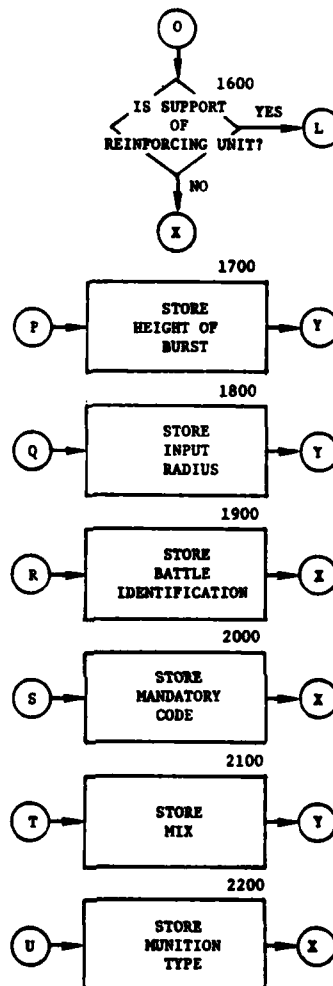


Figure III-2-B-14. Routine KRAKMD (Continued)

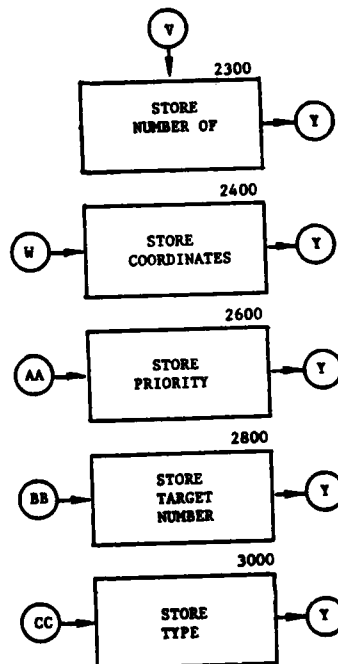


Figure III-2-B-14. Routine KRAKMD (Continued)

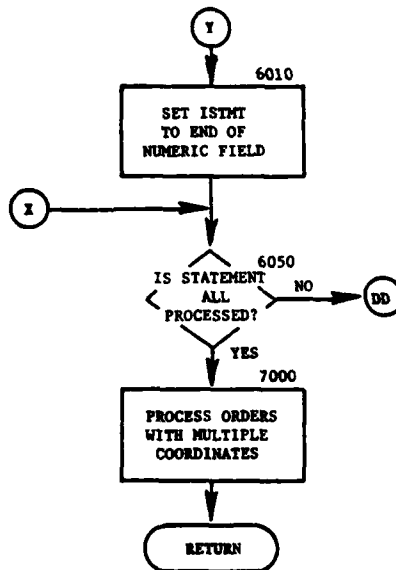


Figure III-2-B-14. Routine KRAKMD (Concluded)

(3) Block L5200. The STMT array is scanned from ISTMT to MXC to determine if a modifier is in that character string. If none is found, a message is printed and the remaining statement is processed. If a modifier is found, IM is set to its index and is used to determine the next block of code to be processed.

(4) Block L100. The aircraft item code is stored in the tenth position in the order array (OARY) and checked to determine if it is less than or equal to 200. An error message is printed if it is not. Control is transferred to block L6010.

(5) Block L200. The altitude is stored in position four in the order array (OARY) and control is passed to block L6010.

(6) Block L300. The speed is stored in the third position of OARY and control is transferred to block L6010.

(7) Block L400. Routine TIME is called to store the time in OARY(2) and control passes to block L6050.

(8) Block L500. The width is stored in OARY(19) and control passes to block L6010.

(9) Block L600. OARY(3) is set to one if a bridge or facility is the modifier and remains zero if a barrier is the modifier. The barrier code is stored in OARY(4) and OARY(5) and checked to determine if the last three characters are numeric. If they are not, an error message is written. Control passes to block L6010.

(10) Block L700. The begin-by code of one is stored in OARY(7) and control is transferred to block L1000.

(11) Block L900. The travel mode mnemonic or reconnaissance by code is stored in OARY(17). The mnemonic or code is checked to determine that it is legitimate and an error message is printed if it is not. Control passes to block L6050.

(12) Block L1000. Write an error message if the time data are missing; otherwise, call TIME to store the begin-by or complete-by time in OARY(9). Transfer control to block L6050.

(13) Block L100. Store depth in OARY(20) and pass control to block L6010.

(14) Block L1200. Store the value of zero in OARY(8) to indicate a desired engineer task and pass control to block L6050.

(15) Block L1300. Store, in OARY(8), a one if the modifier is direct support, two if modifier is reinforcing, or three if modifier is general support-reinforcing. Transfer control to block L3100.

(16) Block L3100. Store the unit identification in OARY(11) and OARY(12); call CHCKID to determine if it is legitimate, and transfer control to block L6050.

(17) Block L1500. Store the time in OARY(2) if the modifier is until; otherwise, add the beginning of the period time (PERBG) to the time and store the negative of the sum in OARY(2). Pass control to block L6050.

(18) Block L1600. If the modifier is general support-reinforcing unit, transfer control to block L1300; otherwise, control is transferred to block L6050.

(19) Block L1700. Store height of burst in OARY(10), write an error message if its value is greater than four, and transfer control to block L6010.

(20) Block L1800. Store impact radius in OARY(9), and pass control to block L6010.

(21) Block L1900. Store the battle identification in OARY(11) and OARY(14), and pass control to block L6050.

(22) Block L2000. Store the value of one in OARY(8) to indicate a mandatory engineer task and transfer control to block L6050.

(23) Block L2100. Store mix in OARY(10), write an error message if its value is greater than 10, and transfer control to block L6010.

(24) Block L2200. Store the munition type in OARY(8), write an error message if its first character is not A, D, or N, and pass control to block L6050.

(25) Block L2300. Determine if the remainder of the number-of modifier is aircraft, escorts, rounds, volleys, or trips. Store number of aircraft in OARY(9), number of escorts in OARY(12), number of rounds or volleys in OARY(7) with number of rounds being stored as the negative value, and number of trips in OARY(9) as a negative value. Transfer control to block L6010.

(26) Block L2400. Store the X coordinate in OARY(5) and the Y coordinate in OARY(6) if it is an on modifier and in OARY(15) and OARY(16) otherwise. If the modifier is either over or to, scan the statement for other coordinate pairs. A check is made to assure that all coordinates are written in pairs, the location defined is in the defined game area, and the coordinates are composed of seven digits. An appropriate message is written if any condition is not met. Control passes to block L6010.

(27) Block L2600. The priority code is stored in OARY(6) if the order is an engineering order and in OARY(18) otherwise. Control passes to block L6010.

(28) Block L2800. The target number is stored in OARY(12) and control is passed to block L6010.

(29) Block L3000. The type is stored in OARY(8), an error message is written if it is not alphabetic, and control is transferred to block L6010.



(30) Block L6010. ISTMT is moved to point to the last numeric character in the data just processed.

(31) Block L6050. If any of the statement remains to be processed, control passes to block L5100.

(32) Block L7000. Orders that have multiple coordinate pairs input are expanded so that only one coordinate pair appears in each order in the final array. The other modifiers are copied into each of these orders. Reconnaissance orders are limited to seven pairs and airmobile assault orders are allowed only four pairs. A message is written if these limits are exceeded. OARY(5) is set to one in the order containing the last coordinate pair for the reconnaissance order. OARY(2) is set to negative one for all airmobile assault orders except the last order.

#### 20. ROUTINE STOW:

a. Purpose. This routine stores the unit scenario or battle paragraph orders in the DSL order file and the number of orders in the unit battle table (UBT) array.

b. Input Variables. DSL common variables UBFLG, UNTPT, BATPT, LABEL, OARY, JOA, NFERR, and ORDFIL.

c. Output Variables. DSL common variables. UBT, NFERR, and RECPT.

d. Logical Flow (Figure III-2-B-15):

(1) Block 1. The variable UPFLG is checked to determine if a unit scenario or battle paragraph is to be stored. If UPFLG is not equal to one indicating it is not a unit scenario, control is transferred to block L2000; otherwise, block 2 is executed.

(2) Block 2. If a unit scenario has been processed, the labels associated with the orders are stored on the DSL data file, and the number of orders in the scenario is stored in UBT.

(3) Block L2000. The number of orders associated with the battle is stored in the UBT array.

(4) Block L2100. The unit scenario or battle paragraph orders stored in OARY are put on to the DSL data file and the record pointer (RECPT) is updated.

#### 21. ROUTINE PASS2:

a. Purpose. This routine processes the second pass necessary to complete the compilation. During this pass, the alphanumeric labels stored by the first pass of the compiler, are converted to the numeric position of the order associated with the label. Some error checks are also made by the routine.

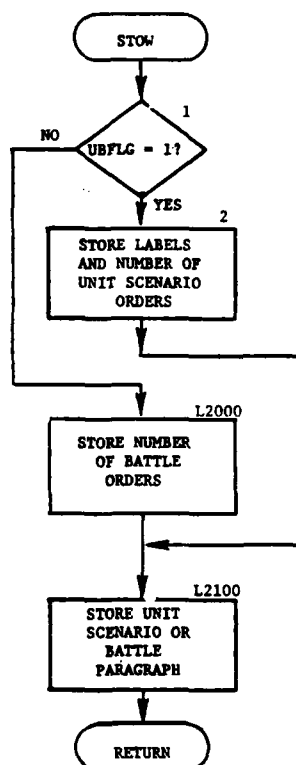


Figure III-2-B-15. Routine STOW

b. Input Variables. DSL common variables. NFERR, NWERR, UBT, UNTPT, BATPT, OARY, LABEL, ORDFIL, and OUT.

c. Output Variables. DSL common variables NFERR and NWERR.

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
OARY	DFORDFIL	Unit scenario or battle paragraph orders.

d. Logical Flow (Figure III-2-B-16):

(1) Block 1. If UNTPT is less than one, no unit scenarios were processed and control is transferred to block L1450.

(2) Block 2. The next unit scenario and its labels are retrieved from the DSL data file. Each order is checked to determine if it contains a label. If it does, the label array is searched for a match. If a match is found, the alphanumeric value of the label in the order is replaced by its numeric position in the label array. If the label is not in the label array, an error message is printed.

(3) Block 3. The engage order is preceded by an advance order. If another relationship occurs, an error message is printed.

(4) Block L1270. Each order is checked to determine if it contains a battle identification. If it does, CHCKID is called to determine if a battle paragraph was input for a battle of that name.

(5) Block L1300. The processed order array is restored in the DSL order file.

(6) Block L1400. If another unit scenario is to be processed, control is transferred to block 2; otherwise, block L1450 is executed.

(7) Block L1450. The battle paragraph of the next battle to be processed is read and the orders are scanned to determine those that contain conditionals or unit scenario labels. The locations of orders containing labels are stored in LABEL.

(8) Block L2100. The labels associated with each unit participating in the battle are retrieved from the DSL data file. The alphanumeric labels stored in the battle paragraph are matched with a label in the array; and, if a match is found, the location of the label is stored in the battle paragraph. An error message is written if the label cannot be matched.

(9) Block L2400. If another participating unit is to be processed, control is transferred to block L2100; otherwise, block L2500 is executed.

(10) Block L2500. If another battle is to be processed, execute block L1450; otherwise, return control to the calling routine.

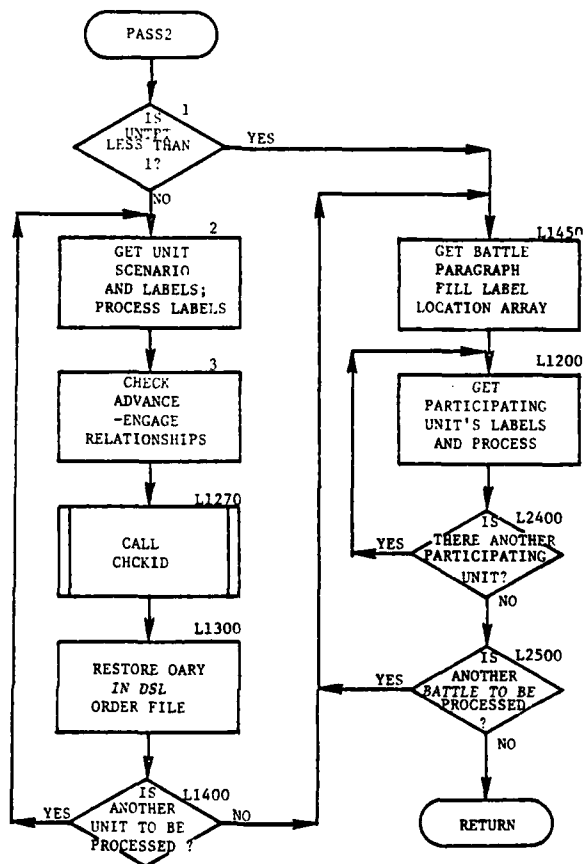


Figure III-2-B-16. Routine PASS2

## 22. ROUTINE DUMPF:

a. Purpose. This routine is designed to give a formatted dump of the DSL data file. This dump is optional at the time the data are compiled and is obtained by specifying the DEBUG option on the DSL card.

### b. Input Variables:

(1) DSL Common Variables. ORDFIL and OUT.

(2) Data File Variables. PERDAY, PERHR, PERMIN, UBT, UNTPT, BATPT, OARY, JOA, LABEL, RECPT, and the length of period.

c. Output Variables. The input variables listed above are output to the line printer.

### d. Processing Description:

(1) The first portion of the routine calls PAGE to eject the page and write the page title. PAGE writes the time the period begins, the length of period, start of game/start of period flag, UNTPT and BATPT. A description of the formats used to write the engineer orders, airmobile orders, conditionals, and other orders is written.

(2) The next portion of the routine reads each unit scenario and its labels, scans each order in the scenario, and prints the order and the appropriate label using a format described above.

(3) The last portion of the routine reads the orders for each battle paragraph, prints the list of participating units, and prints the conditional and labels list with appropriate formats.

## 23. ROUTINE PAGE:

a. Purpose. This routine writes a new DSL period page header and page number at the top of each new DSL compilation page.

b. Input Variables. DSL common block variables LNCNT, MXLN, PGNO, PERDAY, PERHR, PERMIN, PERED, PEREH, and PEREM.

c. Output Variables. DSL common block variables. PGNO and LNCNT.

### d. Logocical Flow (Figure III-2-B-17):

(1) Block 1. If the line counter (LNCNT) is less than the maximum number of lines (MXLN), control returns to the calling routine, otherwise, block 2 is executed.

(2) Block 2. The page number (PGNO) is incremented by one and the line counter (LNCNT) is reset to zero.

(3) Block 3. The DSL compilation page header and page number is written.

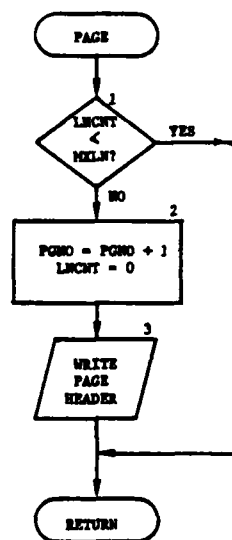


Figure III-2-B-17. Routine PAGE

III-2-B-52

#### 24. ROUTINE TIME:

a. Purpose. This routine converts the character time expressed in days, hours, and minutes, or as six digits, to an integer time in centiminutes.

b. Input Variables:

(1) DSL Common Block Variables. ISTMT, STMT, IALPH, PERBG and NFERR.

(2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
LOC	Call	Pointer to the first character following the first numeric field.
IF	Call	Fatal error test indicator: IF = 0, do not check, less than PERBG IF = 1, check, less than PERBG.
IVAL	Call	Integer value of first numeric character.

c. Output Variables:

(1) DSL Common Block Variables. NFERR and ISTMT.

(2) Other Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
LOC	Call	POINTER to second character following last numeric field.
ITIME	Call	Time in centiminutes.

d. Logical Flow (Figure III-2-3-18):

(1) Block L31. Determine if the characters following the first numeric value are DA, HO, HR, or MI. If they are, the corresponding value (ID, IH, or IM) is set equal to IVAL; otherwise, the time is assumed to be in the six-digit format and block L1 is executed.

(2) Block L1. IVAL is divided into day, hour, and minute values. The day, hour, and minute values are converted to centiminutes and summed.

(3) Block 2. FINDN is called and returns the location of the next numeric character.

(4) Block 3. IBCD is called and returns the integer value of the numeric characters found by FINDN.

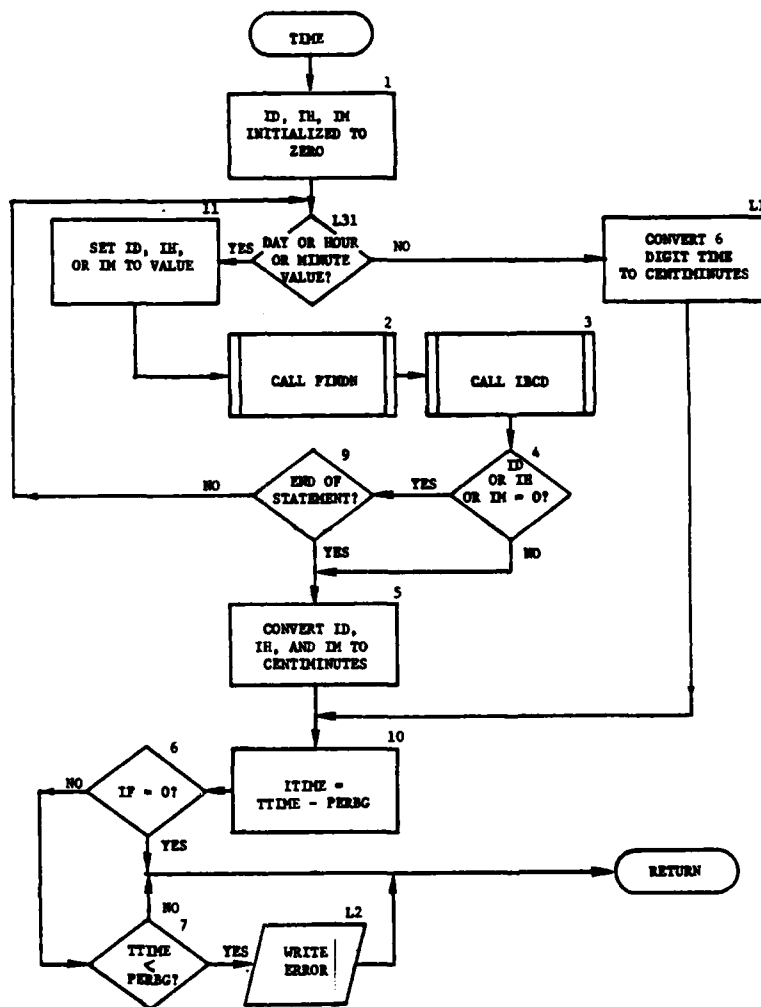


Figure III-2-B-18. Routine TIME

III-2-B-54



(5) Block 4. The values of ID, IH, and IM are checked. If any value is equal to zero, block 9 is executed; otherwise, control goes to block 5.

(6) Block 9. If the end of the statement has not been reached, execute block L31.

(7) Block 5. The day, hour, and minute values (ID, IH, and IM) are converted to centiminutes and summed.

(8) Block 10. The time the period began (in centiminutes) is subtracted from the time calculated and the result is stored in ITIME.

(9) Block 6. The value of IF is checked. If IF is nonzero, block 7 is executed; otherwise, control is returned to the calling routine.

(10) Block 7. If the time in centiminutes is less than PERBG, block L2 is executed; otherwise, control is returned to the calling routine.

(11) Block L2. The number of fatal errors (NFERR) is incremented by one, and an error message stating that the time is prior to the beginning of the period is written.

#### 25. ROUTINE CHCKID:

a. Purpose. This routine checks the unit identification to determine if it begins with B or R and contains eight characters. It also checks to determine if the unit exists, is a resolution unit, and is not a duplicate unit. Battle identifications are checked to determine if any name contains more than eight characters and if the battle is duplicated.

##### b. Input Variables.

(1) DSL Common Block Variables. UNTPT, BATPT, IBLNK, UBT, UBFLG, UBID, NFERR, and NWERR.

(2) Common TWO Variable. UNTLOC.

(3) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IFLG	Call	Flag indicating type of checking requested on UB. 0 = UID - check for duplication check for eight characters check for B or R in first character check for resolution unit.  BID - check for duplication.

<u>Name</u>	<u>Source</u>	<u>Contents</u>
		1 = check UID only for 8 characters and B or R in first character. 2 = check to see if BID is in UBT.
UB	Call	Containing unit or battle identification to be checked (double precision).

c. Output Variables.

- (1) DSL Common Block Variables. NFERR, NWERR.
- (2) Other Variables. None.

d. Logical Flow (Figure III-2-B-19):

(1) Block 1. The value of IFLG determines the type of check to be made. If IFLG is equal to zero, block L19 is executed. Block L2 is executed if IFLG is equal to one and block 2 is executed if IFLG is equal to two.

(2) Block L2. The unit identification is checked to verify that it contains eight characters. Block L8 is executed if the unit identification does not contain eight characters.

(3) Block L91. The first character of the unit identification is checked. Block L9 is executed if the first character is not B or R.

(4) Block 3. If IFLG is equal to one, control is returned to the calling routine. Block 4 is executed if IFLG is not equal to one.

(5) Block 4. Routine IUIDF returns the identification number of a unit. Block 5 is executed if the subscript is greater than one and block L15 is executed if the subscript is equal to zero.

(6) Block 5. If the unit's location is equal to zero, block L10 is executed. Control is returned to the calling routine otherwise.

(7) Block L19. Block L22 is executed if UBFLG is equal to one; otherwise, block L21 is executed.

(8) Block L22. UBID is checked to determine if the unit identification is a duplicate. Block L7 is executed if the unit is a duplicate; otherwise, block L2 is executed.

(9) Block L21. UBID is checked to determine if the battle identification is a duplicate. Block L4 is executed if it is a duplicate, or block 2 is executed if it is not.

(10) Block 2. If the ninth character of the battle identification is not a blank, the identification is greater than eight characters and block L14 is executed. If it is a blank, control is returned to the calling routine.



(11) Block L8. The number of fatal errors (NFERR) is incremented by one, and an error message is written stating the unit identification contains fewer than eight characters.

(12) Block L9. The number of fatal errors (NFERR) is incremented by one, and an error message is written stating the unit identification does not begin with B or R.

(13) Block L15. The number of warning errors (NWERR) is incremented by one, and an error message is written stating the unit does not exist.

(14) Block L10. The number of warning errors (NWERR) is incremented by one, and an error message is written stating the unit is not a resolution unit.

(15) Block L7. The number of fatal errors (NFERR) is incremented by one, and an error message is written stating the unit is a duplicate.

(16) Block L4. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written stating the battle is a duplicate.

(17) Block L14. The number of fatal errors (NFERR) is incremented by one, and a fatal error message is written stating the battle identification contains more than eight characters.

#### 26. ROUTINE IUIDF:

a. Purpose. This routine scans the list of units (UIDTAB) and returns the location of the unit identification input to it.

##### b. Input Variables:

- (1) Standard Common Block Variables. UIDTAB, BPOINT, and RPOINT.
- (2) Other Variables.

<u>Name</u>	<u>Source</u>	<u>Description</u>
NUID	Call	Unit identification to be matched.

##### c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Description</u>
IUIDF	Call	Location in UIDTAB of NUID.

d. Processing Description. The routine determines if the unit is Blue or Red force. Using this information it initiates a search loop from one to BPOINT or RPOINT to 1000. If the unit identification input is matched in the list of units (UIDTAB), its location is returned in IUIDF; otherwise, a zero is returned. If the unit identification input (NUID) has a first character that is neither B nor R, an error message is printed and a zero is returned

27. ROUTINE MATCH:

a. Purpose. This routine searches for a string of characters in STMT that matches a string of characters in MDFX.

b. Input Variables:

(1) DSL Common Block Variables. STMT, ISTMT, IBLNK, and IPEP.

(2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
MDFX	Call	Array of character strings to be matched (one character per word, left justified, blank filled).
NM	Call	Number of character strings.
MXC	Call	Index of STMT where search will end.

c. Output Variables:

(1) DSL Common Block Variable. ISTMT (pointing to the last character processed).

(2) Other Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
IFND	Call	Pointer to the string matched: 0 = no match was found 1 = a period was encountered.

d. Logical Flow (Figure III-2-B-20):

(1) Block L1000. The statement pointer (ISTMT) is incremented by one so it points to the character being analyzed.

(2) Blocks 1 and L7000. A check determines if the character being analyzed is a period. If it is, block L7000 sets IFND equal to negative one.

(3) Blocks 2 and 3. It is determined whether the character being analyzed is equal to the first character of a string in MDFX. If it is not equal, the index of MDFX is incremented to the next string.

(4) Blocks 4, L2010, and 5. A check determines if the next three characters in STMT are equal to the next three characters of the string in MDFX. If they are equal, IFND is set equal to the pointer of the string matched in MDFX and ISTMT is set equal to the pointer to the last character processed.

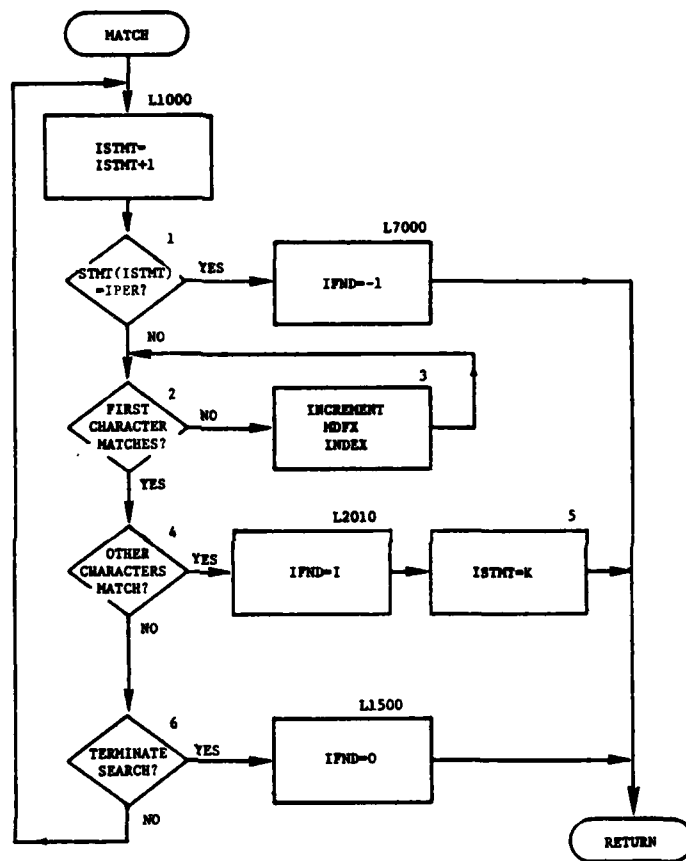


Figure III-2-B-20. Routine MATCH

III-2-B-60

(5) Blocks 6 and L1500. A check determines if the search is terminated (MXC -1 > ISTMT). If so, IFND is set equal to zero.

28. ROUTINE FINDN:

a. Purpose. This routine locates the next numeric character in a string of characters and returns a constant containing the number of nonnumeric characters encountered before a numeric was encountered.

b. Input Variables:

(1) DSL Common Block Variables. IPER and INGR.

(2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
S	Call	Array containing the character string.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
N	Call	Number of nonnumeric characters encountered before a numeric was encountered.

d. Logical Flow (Figure III-2-B-21):

(1) Block 1. Character S(N) is checked to determine if it is a period. If S(N) is a period, block 3 is executed; if not, block 2 is executed.

(2) Block 2. If S(N) is numeric, block 4 is executed; if not, block 5 is executed.

(3) Block 3. The value of N is set to negative one.

(4) Block 4. One is subtracted from the value of N.

(5) Block 5. N is incremented by one.

29. ROUTINE EXPAND:

a. Purpose. This routine expands and unpacks the number of character strings that are stored in the array M to the array MX.

b. Input Variables.

<u>Name</u>	<u>Source</u>	<u>Contents</u>
M	Call	Array containing original character strings four characters per word, left justified, and blank filled.
N	Call	Number of character strings.

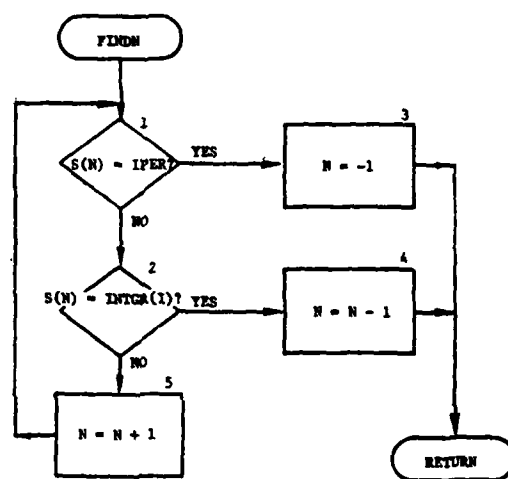


Figure III-2-B-21. Routine FINDN

III-2-B-62



c. Output Variables.

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
MX	Call	Array containing character strings (one character per word).

d. Processing Description: EXPAND calls MVCHAR four times for each string and stores one character in one word of MX during each call.

30. ROUTINE IBCD:

a. Purpose. This routine converts numeric characters to an integer value and returns the integer value, the number of characters converted, and the statement pointer.

b. Input Variables:

(1) DSL Common Block Variables. STMT and INTGR.

(2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ISTM	Call	Pointer to the first numeric character to be converted in STMT.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
ISTM	Call	Pointer to first nonnumeric character encountered.
NC	Call	Number of numeric characters encountered.
IR	Call	Integer value of converted characters.

d. Processing Description. NC and IR are initialized to zero. Each numeric character encountered is converted by multiplying the previous value of IR by 10 and adding the value of the new character. Each time a numeric is encountered NC and ISTM are incremented by one.

31. ROUTINE RBCD:

a. Purpose. This routine converts numeric characters to real values and returns the real value, the number of characters converted, and the statement pointer.

b. Input Variables:

(1) DSL Common Block Variables. STMT and INTGR.

(2) Other Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ISTM	Call	Pointer to the first numeric character to be converted in STMT.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
ISTM	Call	Pointer to first nonnumeric character encountered.
NC	Call	Number of numeric characters encountered.
R	Call	Real value of converted characters.

d. Processing Description. NC and R are initialized to zero. Each numeric character encountered is converted by multiplying the previous value of R by 10 and adding the value of the new character. Each time a numeric is encountered NC and ISTM are incremented by one.

## APPENDIX C

### DIVWAG SCENARIO LANGUAGE COMPILER OUTPUT DESCRIPTIONS

1. INTRODUCTION. This appendix contains samples and detailed descriptions of printed output from routines within the DIVWAG Scenario Language (DSL) Compiler of the Orders Input Processor. A figure depicts the format of each routine's print statements. In the figure, an alphabetical character (descriptor) designates an appropriate line, group of lines, or column that is explained in the following paragraphs.
2. ROUTINE DSLINT. This printed output (Figure III-2-C-1) is a listing of the DIVWAG file name table. Column 1 lists the number of the file; column 2 contains the starting address of the file; column 3 contains the number of words per record; and column 4 contains the number of records in the file.
3. ROUTINE RDSTMT. Routine RDSTMT lists the input cards as they are read. Figure III-2-C-2 illustrates several different kinds of input cards:

<u>Output Descriptor</u>	<u>Explanation</u>
A	The DSL control cards are listed in this group. The first card contains the debug option; the second card indicates this period is to start at day 1, hour 5, and minute 0; and the third card sets the period length to 150 minutes.
B	This printout contains examples of unit scenarios. The Blue force unit, B1011ZAR, has two MOVE orders followed by a PREPARE order. The Red force unit, R1330ZTK, has a PREPARE order contingent upon a conditional, a pseudo order GOTO also contingent upon a conditional, an ADVANCE order, an ENGAGE order, and a second PREPARE order. Its scenario also contains a COMMENT and two labeled order.
C	The battle MECH3 will involve the units R2110ZTK, R2120ZTK, R2130ZTK, and B8001ZAR which will execute orders labeled M3, M3, M3, and M1 in their respective unit scenarios when game time exceeds day 1, hour 5, and 14 minutes.

4. ROUTINE PAGE. This routine positions the printer at the top of the next page and prints the header shown in Figure III-2-C-3.
5. ROUTINE PASS2. This routine prints the following message:

#### PASS2 DIAGNOSTICS

It is printed at the top of the page as an indication that the routine has been entered and to separate any diagnostic print from the listing of the data input cards.

		FNT	
1	27393	512	1000
2	2558442	35	4250
3	1213610	10	15001
4	893273	9	3133
5	0	0	0
6	556273	10	400
7	1930483	300	52
8	1353620	3200	1
9	1975183	1000	5
10	0	0	0
11	976222	512	249
12	2149842	35	3000
13	1355820	401	401
14	1984583	20	950
15	1981383	200	16
16	2129842	20000	1
17	1103710	700	157
18	2462442	6	1000
19	2003583	120	50
20	956222	20	1000
21	539393	256	105
22	2707192	752	20
23	2054376	25	200
24	1946083	1212	25
25	257	27092	1
26	1527621	72	616
27	1571973	112	1000
28	2009583	189	237
29	920570	215	72
30	2722232	2149	23
31	2059376	10	4195
32	0	0	0
33	0	0	0
34	0	0	0
35	0	0	0
36	936122	20100	1
37	2468442	60	1500
38	0	0	0
39	2101326	14255	2
40	0	0	0
41	1824043	340	84
42	1852603	680	21
43	1856883	3400	4
44	1880483	100	500
45	1683973	35	4002
46	0	0	0
47	0	0	0
48	2254842	1212	50
49	0	0	0
50	570273	202	1000
51	772273	1000	2
52	774273	202	500
53	875273	24	750
54	0	0	0
55	2315442	21	7000
56	0	0	30

Figure III-2-C-1. Routine DSLINT Printed Sample Output

III-2-C-2

(A) [ DSL,DEBUC.  
 START OF GAMEX 01 DAY 05 HOUR 00 MIN.  
 PERIOD LENGTH: 150 MIN.

(B) [ IDXR10117AP.  
 MOVE TO 0099300-0099300,0100300-0099300,0102100-0099300,0104200-0099300,  
 0106900-0099300,0108300-0099300,0109400-0099300,0111900-0099300 BY TPOH.  
 MOVE TO 0114700-0099300,0116300-0099300,0118400-0099300,0120300-0099300,  
 BY TPOH.  
 PREPARE UNTIL 010731 AT WIDTH 2000-DEPTH 2000.

(B) [ IDXR1330ZTK.  
 COMMENTX 30 TK BY 30 TKR 1ST TKO / MSN ATTACK.  
 M2IF TIME GREATER THAN 010514, THEN PREPARE FOR 15 MINUTES  
 AT WIDTH 2100-DEPTH 2100.  
 IF R1330ZTK EQUIPMENT TYPE IS LESS THAN 20, THEN GO TO PR.  
 ADVANCE TO 211300-0086000 BY TCCR.  
 ENGAGE IN BATTLE MECW2.  
 PRXPREPARE UNTIL 010731 AT WIDTH 2100-DEPTH 2100.

(C) [ BATTLEXMECW3.  
 SURFACE UNITSXR2110ZTK,R2120ZTK,R2130ZTK,98001ZAP.  
 WHEN TIME GREATER THAN 010514, THEN M3,M3,M3,M1.

Figure III-2-C-2. Routine RDSTMT Printed Sample Output

7 S L C O M P I L A T I O N PAGE 7  
 PERIOD STARTS 1 DAY 5 HOUR 0 MINUTES ENDS 1 DAY 7 HOUR 30 MINUTES.

Figure III-2-C-3. Routine PAGE Printed Sample Output

6. ROUTINE DUMPF. This routine gives a formatted dump (Figure III-2-C-4) of the tables generated by the DSL compiler if the DEBUG parameter is specified on the first DSL control card.

<u>Output Descriptor</u>	<u>Explanation</u>
A	The banner line is printed and the day, hour, and minute the period begins and the length of the period are listed.
B	The period type and an explanation of its meaning is listed on the first line. The following lines give the pointers to the last positions used in the unit battle table.
C	All order information is not stored in the same format. Descriptor C lists the various formats used to print the unit scenario orders shown in descriptors F and G. The first column printed by each format (JOA) is the counter indicating the relative record the order is placed in. Column two contains the label associated with the order. If the order was not labeled, that space will be blank. The third column contains the order number (referred to as NORD in the Period Processor), if the value is greater than zero. If the value is a -1 the pseudo order GOTO is indicated. If the value is less than -1 a conditional is stored in that record. A description of the remaining columns is tabulated below for each format type.

Engineer orders, NORD = 42, 43, 44, 45

BARR-BRID-FAC CODE: code indicating type of engineering structure; 0 = barrier, 1 = bridge or facility

BID(1): first half of barrier-facility name

BID(2): second half of barrier-facility name

PRIORITY: priority of task

TIME FLAG: 0 = complete by, 1 = begin by

DESIR/MAND: 0 = desired, 1 = mandatory

ENG TIME: time in centiminutes

Airmobile orders, NORD = 20, 39, 40, 41

AT TIME: time associated with order

NO. OF A/C, NO. TRIPS: value greater than 0 indicates number of trips

MIX: transport-escort mix code or aircraft item code

NO. OF ESCORTS: number of aircraft or target number

OBJX,OBJY: coordinates of mission destination

ORDPROFILE 0JND FPOW ROUTINE JUMPF

DAY PERIOD BEGINS 1  
 HOUR PERIOD BEGINS 5  
 MINUTE PERIOD BEGINS 9

LENGTH OF PERIOD IN CENTIMINUTES 15000

PERIOD TYPE 7 1= NORMAL PERIOD, 3= FIRST PERIOD OF GAME  
 UNTOT= 40 POINTED TO THE LAST UNIT LOADED  
 BATPT= 1019 POINTER TO THE LAST BATTLE LOADED

FORMAT FOR ENGINEER ORDERS WITH NORO=(42,43,44,45)

JOA,LABEL,NORO,PARR-ROJD-FAC CODE,RI011,91012,PRIORITY,TIME FLAG, DESIR/MANO, ENG TIME

FORMAT FOR AIRMOBILE ORDERS WITH NORO=(20,39,40,41)

JOA,LABEL,NORO,AT TIME,NO,AVC>J,NO,TRIPSI3,MIX, NO. OF ESCORTS,09JX,09JY

FORMAT FOR TRUTH FUNCTION ORDERS,NOROI-1

JOA,LABEL,NORO,TFCD,TUI011,TUI012,CITIME,XLOC,YLOC,AVAL,PVAL,NOT,L07,IFC  
 POINTER-FALSE,POINTER-TRUE

FORMAT FOR NORMAL ORDERS NORO=ANYTHING NOT PREVIOUSLY SPECIFIED

JOA,LABEL,NORO,WTIME,AIRSPD,ALT,06ZX,06ZY,NOR,MUNTPY,I3,N08,CONOUI0,3ATI0,09JX,09JY,002CD,NOVPOIY,  
 BLANK,BLANK,POINTER

Figure III-2-C-4. Routine DUMPF Printed Sample Output  
 (Continued on Next Page)

	UID	PCOUNTER AND NUMO	URECNO
1	BA0127MT	1	202
2	BA0112MT	1	208
3	BA0027AP	1	214
4	BA0017AR	1	220
5	B10117AR	13	226
6	B10217AP	15	244
7	B10317MT	13	264
8	B10107CA	11	282
9	B1013CAC	1	298
10	B10237AM	2	304
11	B101AWPL	1	311
12	B10AV7BN	1	317
13	B10137AC	12	323
14	B1100FHO	4	340
15	B1200FHO	4	349
16	B13ACPHO	5	354
17	BFAIRSUB	2	368
18	B1AIRFLT	6	375
19	BFAIRFLT	7	386
20	B1AFRMRP	2	391
21	B1AFRCF	5	405
22	BFAIRFCF	5	415
23	BFAIRFCF	1	425
24	B1010PCN	1	431
25	P13107TK	7	437
26	P13207TK	7	449
27	P13307TK	7	461
28	P14307MT	1	473
29	P14407TK	2	479
30	P7010RCN	1	486
31	P21107TK	7	492
32	P21207TK	7	504
33	P21307TK	7	516
34	P24407TK	1	528
35	P2410ZMT	1	534
36	R10000PS	8	540
37	P20000PS	8	553
38	P1AIRPCN	4	566
39	P2AIRPCN	5	575
40	PFAIRFT	2	585

	BIT	NUMO AND NUMT	BRECNO
1	LONG	7024	611
2	MECH3	6004	605
3	MECH2	6002	599
4	MECH1A	6002	593
5	MECH1	6002	587

Figure III-2-C-4. Routine DUMPF Printed Sample Output (Continued)



OFDEPS FOR UNIT 912112AR UNIT NO. 5												
1	3	0	3	0	0.	0.	0	0	0	98300.	99300.	TRDM
2	3	0	0	0	0.	0.	0	0	0	100300.	99000.	TRDM
3	3	0	3	3	0.	3.	0	0	0	102100.	98200.	TRDM
4	3	0	0	0	0.	0.	0	0	0	104200.	97900.	TRDM
5	3	0	3	0	0.	0.	0	0	0	106900.	96700.	TRDM
6	3	0	0	0	0.	0.	0	0	0	108300.	96000.	TRDM
7	3	0	0	0	0.	0.	0	0	0	109400.	95100.	TRDM
8	3	0	0	0	0.	0.	0	0	0	111900.	95000.	TRDM
9	3	0	0	0	0.	0.	0	0	0	114700.	95300.	TRDM
10	3	0	0	0	0.	0.	0	0	0	116300.	96200.	TRDM
11	3	0	0	0	0.	0.	0	0	0	118400.	95600.	TRDM
12	3	0	0	0	0.	0.	0	0	0	120000.	94700.	TRDM
13	2	45100	0	0	0.	0.	0	0	0	0	2000	0

Figure III-2-C-4. Routine DUMPF Printed Sample Output (Continued)

[illegible]

BATTLE NO.	BATTLE NO.		BATTLE NO.
	1	2	
5	1	1	1400
6	1	1	1400

Figure III-2-C-4. Routine DUMPF Printed Sample Output (Concluded)

Output  
Descriptor

Explanation

Conditionals, value of column 3 is less than zero  
TFCD: conditional code - printed in column 3  
TUID(1): first word of UID  
TUID(2): second word of UID  
CTIME: time associated with conditional  
XLOC,YLOC: X,Y coordinates associated with conditional  
AVAL: amount of equipment item associated with conditional  
PVAL: percent of equipment item associated with conditional  
NOT: value is true if negator appears in conditional  
LOG: logical variable indicator; 1 = less than, 0 = equal  
to, +1 = greater than  
IEC: equipment item code

Other orders

NTIME: time in centiminutes, negative if data were from  
FOR modifier  
AIRSPD: air speed  
ALT: altitude  
DGZX,DGZY: designated ground zero X and Y coordinates  
NOR: number of rounds  
MUNTPY: munition type  
IR: impact radius  
HOB: height of burst  
COOPUID: UID of cooperating unit  
BATID: battle identification  
OBJX,OBJY: coordinates of destination  
RRRCD: travel mode mnemonic or reconnaissance by code  
MOVPRTY: priority of movement

The last two columns printed are reserved for pointers to  
other orders; they contain nonzero values for all conditionals,  
and the last column will be nonzero for the pseudo order GOTO.

D Descriptor D lists the unit portion of the unit battle table.  
Column 1 numbers the units as they are listed and also serves  
as a page index to the following printout. Column 2 contains  
the unit identification; column 3 contains integers indicating  
the number of DSL data file records required to store the unit  
scenario; and the fourth column contains the number of the first  
record used to store the scenario.

E The format of the battle portion of the unit battle table  
listed in group E is similar to that of the unit portion.  
Again, column 1 numbers the battles and serves as a page in-  
dex to that portion of the following printout. Column 2 con-  
tains the identification of the battle. Column 3 contains

Output  
Descriptors

Explanation

two variables packed in each word. The rightmost three digits indicate the number of units involved in the battle, and the remaining digits denote the number of records required to store the battle paragraph information. Column 4 contains the record number of the first record used to store that battle paragraph.

- F Group F is a formatted dump of the table generated from the orders contained in the unit scenario of unit B1011ZAR. It is unit number 5 in the directory table listed in descriptor D and it's unit scenario is listed in RDSTMT (descriptor group B, Figure III-2-C-2). As indicated by the last print format described in descriptor C, the first column is a counter of the records used to store the orders. The second column is blank. The third column gives the order number (NORD). The fourth column contains meaningful information only in the thirteenth record. The value listed is the number of centiminutes from the start of the period to the time input in the PREPARE order. The last three columns contain the X and Y coordinates and the travel mode mnemonic respectively. It should be noted that one record is generated for each coordinate pair in each MOVE order.
- G Group G lists the table generated from the unit scenario of R1330ZTK unit number 27 in the directory. Again, column 1 is the record counter. Column 2 contains the labels of the labeled orders. Column 3 contains order codes. The positive numbers are NORDs and the negative numbers indicate a conditional or a GOTO pseudo order. This column also serves as an index to the format descriptions listed in descriptor group C.
- H This group lists the information stored for battle MECH3. Its battle paragraph is listed in RDSTMT (descriptor C, Figure III-2-C-2). The second line contains a list of the units participating in the battle. The remaining lines list the conditionals and the associated labels. Note that the record count in column one begins with 5; the first four records are reserved for the list of participating units, and record number 5 is the time conditional. The last two columns listed for a conditional contain the record numbers to be transferred to (i.e., if the conditional is false or true respectively). Record 6 now contains the record numbers of the orders that the participating units are to execute if the battle is terminated by this conditional.

APPENDIX D

SOURCE LISTINGS FOR ORDERS INPUT PROCESSOR DIVWAG SCENARIO LANGUAGE COMPILER

(AVAILABLE UNDER SEPARATE COVER)

NOTES

III-2-D-2

## CHAPTER 3

### OPERATING INSTRUCTIONS

1. INTRODUCTION. The Operating Instructions (OPERINS) loader provides a means for the gaming staff to interject, at the beginning of a period of simulated combat, certain items for the control of ground-based sensors, for the introduction of intelligence from sources beyond the scope of the sensors simulated within the DIVWAG Period Processor, and for the control of fire support allocation. Operating Instructions input must be provided for the initial period of a game to initialize ground-based sensors and fire support allocation control and may be provided for any following game period.
2. GROUND-BASED SENSORS. The classes of ground-based sensors for which Operating Instructions are required include moving target indicator (MTI) radars, countermortar and counterbattery (CM/CB) radars, air defense (AD) radars and unattended ground sensor (UGS) fields. Basic hardware characteristics of these classes of sensors are loaded within the Intelligence and Control Model constant data base (Section IV, Chapter 6, Appendix A). Operating Instructions identify individual radars and unit ground sensor fields as to their locations, operational constraints that are expected to vary in the course of a game, and composition for UGS fields.
3. INTRODUCTION OF INTELLIGENCE. Collection of information concerning the enemy force is simulated within the DIVWAG system for selected ground-based and aerial sensors typically available to the committed forces. Explicit simulation of all potential sources of information is not attempted. Through the use of Operating Instructions, the gaming staff may introduce additional information concerning the opposing forces. Such information may be routed through the information processing areas of the Intelligence and Control Model and/or through fire support allocation areas of the model at the discretion of the gaming staff.
4. FIRE SUPPORT ALLOCATION CONTROL. Operating Instructions allow the gaming staff to set the most critical parameter used within the system to control the allocation of fire support and to define the number and type of tactical air sorties available to the respective forces.

NOTES

III-3-2



## APPENDIX A

### OPERATING INSTRUCTIONS DATA INPUT REQUIREMENTS

1. SENSOR CONTROL. The DIVWAG system will simulate up to 100 model sensors for each opposing force where a model sensor is an individual moving target indicator (MTI), countermortar/counterbattery (CM/CB), or air defense (AD) radar or an unattended ground sensor (UGS) field. Each such model sensor is defined to the DIVWAG system through the use of the OPERINS loader. A distinct card format is required for each type sensor as discussed below. Certain data items, including a sensor code, unit identification (UID) of first node, sensor index, sensor reference number, and sensor status code are common to all card formats.

#### a. Data Common to All Sensors:

(1) Sensor Code. Each generic type sensor is identified by a sensor code which appears on all sensor control data cards. Permissible values of this code and interpretations are:

- 02 = short range MTI radar
- 03 = medium range MTI radar
- 04 = long range MTI radar
- 05 = continuous tracking type CM/CB radar
- 06 = dual beam type CM/CB radar
- 07 = UGS field
- 08 = AD radar.

(b) UID of First Node. Each sensing report is passed, within the Intelligence and Control Model, through an intelligence communications network for decisions and processing. (See Section IV, Chapter 6 for a detailed discussion.) Nodes within this network are battalion, brigade, or division level command units. This entry identifies the unit that is nominally in control of the sensor being specified. Each sensor will use the unit so identified as the entry point into the network for all sensing reports generated by that sensor. The UID is the eight-character alphanumeric unit identifier of the unit to be specified. This unit must have a maneuver type unit type designator (UTD); that is, the third character of this unit's UTD must be M.

(c) Sensor Index. If a given unit (UID) is responsible for more than one MTI radar, more than one CM/CB radar, more than one AD radar, or more than one UGS field, the sensor index is used to determine to which radar or UGS field the OPERINS data applies. A sequence of sensor indexes, starting with 1, must be assigned to each group of model sensors that is composed of elements from one of the above categories and shares the same first node UID. For example, if a given unit controlled three MTI radars, one CM/CB radar, and six UGS fields; the sensor indexes applied to the MTI radar would be 1, 2, and 3; to the CM/CB radar 1; and to the UGS fields 1, 2, 3, 4, 5, and 6.

(d) Sensor Reference Number. Hardware characteristics of each type sensor are defined using the Intelligence and Control Model constant data load routines (see Section IV, Chapter 6, Appendix A). These routines assign a reference number to each type sensor. This reference number is used to specify within OPERINS data the piece of hardware desired. The reference number is obtained from the printed output of the Intelligence and Control load routines.

(e) Sensor Status Code. Allowed values of the sensor status code are:

- 1 = available
- 2 = tracking
- 3 = inoperative.

The status code is set dynamically to appropriate values by the Intelligence and Control Model. An entry of 1 using OPERINS data will permit full play of the sensor during the course of the period. An entry of 2 or 3 made by OPERINS is not reset by the Intelligence and Control Model and will effectively turn off the sensor through the course of the period.

b. MTI Radar. The essential hardware characteristics of this type radar are entered in the constant data input. These performance characteristics are not changed during the entire game. The operating instructions, which the gamer is expected to give, are those that deal with operating constraints and changes in the situation as the battle progresses. This card format must be prepared for each sensor played either to define the sensor initially or to change a previously defined sensor's location, status, and orientation. There are two segments in the input card format, illustrated in Figure III-3-A-1. The first segment, columns 1 through 18, identifies a specific radar and associates its operating unit with identification details. The second segment, columns 20 through 66, specifies the location and other operating parameters.

(1) Sensor Code (Columns 1-2). Each generic type sensor is identified by a sensor code. For MTI radar, enter the value 02 (short range), 03 (medium range), or 04 (long range).

(2) UID of Controlling Unit (Columns 4-11). Enter the eight-character UID of the unit that is to receive reports from this sensor. (This must be a unit having an M as the third character of its UTD.)

(3) Unique Sensor Index (Columns 13-14). When the controlling unit is responsible for more than one MTI radar, this index is used to discriminate among the radars. The index must be assigned serially, starting with 1 for the first MTI radar under this unit, 2 for the next MTI radar, etc. In relocating or reorienting a radar, this index is used to decide which radar to move, should a unit be responsible for more than one. The entry is right justified.



(4) Sensor Reference Number (Columns 16-18). Each specific type of sensor hardware is parametrically defined within the constant data input on the Intelligence and Control Model data file. In the process, a reference number is assigned by the data load program. Enter this number, obtained from the load program's printed output.

(5) Sensor Status Code (Column 20). The status of the radar at the beginning of the game or period is to be entered in this card column.

(6) Sensor Coordinates (Columns 22-36). Enter the seven-digit X and Y coordinates of the sensor right adjusted in appropriate columns. Leading zeros are not required.

(7) Assigned Area of Search (Columns 46-66). The area of search to be assigned to an MTI radar is specified by the entries on columns 46-66. Columns 46-50 and 52-56 contain the minimum and maximum ranges, respectively, of the assigned search area, in meters. Columns 58-61 contain the orientation, or central azimuth, of the assigned search sector, measured clockwise in mils from grid north. Columns 63-66 contain the width of the assigned search sector, in mils. These values must not exceed hardware characteristics as loaded in the Intelligence and Control Model constant data.

c. CM/CB Radar. The hardware characteristics of this type radar were entered in the constant data input. These characteristics are not changed during the entire game. The data on countermortar/counterbattery radar loaded as Operating Instruction are those that deal with operating constraints and changes in the situation as the battle progresses. A card in this format (Figure III-3-A-2) must be prepared for each sensor played either to define the sensor initially or to change a previously defined sensor's location status or orientation.

(1) Sensor Code (Column 1-2). Each generic type sensor is identified by a sensor code. For a single beam continuous tracking CM/CB radar, set the code to 5. For a dual beam CM/CB radar, set the code to 6. The entry must be right justified.

(2) UID of First Node in Communication Net (Columns 4-11). Enter the eight-character UID of the unit that will be the first node in a communication network from this specific radar site. This unit must be a maneuver unit (i.e., have the character M in the third character of the UTD).

(3) Unique Sensor Index (Columns 13-14). When this receiving node has more than one CM/CB radar reporting to it, this index is used to discriminate among the radars. The index must be assigned serially, starting with 1 for the first CM/CB radar reporting to this unit, 2 for the second, etc. In relocating or reorienting a radar, this index is used to decide which radar to move or reorient, should the unit have more than one reporting to it. The entry must be right justified.



(4) Sensor Reference Number (Columns 16-18). Each specific type of sensor hardware is parametrically defined within the constant data input on the Intelligence and Control Model data file. In the process, a reference number is assigned by the data load program, and it is this number that is required here to tie this particular radar to a set of hardware data.

(5) Sensor Status Code (Column 20). The status of the radar at the beginning of the game or period is to be entered in this card column. Entering a 2 or 3 will preclude the radar from detecting and recognizing targets during the course of the period. Entering a 1 will permit full play of the radar during the course of the period.

(6) Sensor Coordinates (Columns 22-36). Enter the seven-digit X and Y coordinates of the sensor. The entry must be right justified, and leading zeros are not required.

(7) Dual Beam Status (Column 44). This is a status code for dual beam CM/CB radars only (i.e., sensor code 6). This flag is used to indicate the desired operational status of the radar. Possible entries are:

- 0 = monitor low angle fire only
- 1 = monitor high angle fire only
- 2 = switch back and forth to monitoring  
low and high angle fire.

The radar must have this capability.

(8) Center Azimuth,  $\alpha$  (Columns 58-61). This the orientation or center azimuth of the search sector, measured in mils from grid north. Entry must be right justified.

(9) Search Sector,  $\theta$  (Columns 63-66). This is the assigned search sector width, in mils. If this search sector is more than two times the sector coverage capability of the hardware (see beam description variables in Appendix A to Chapter 6 of the Period Processor, Section IV for Countermortar/Counterbattery Radars), there will be a coverage gap in the center of the assigned search sector. Entry must be right justified.

(10) Site Mask Angle,  $\beta$  (Columns 68-71). This is the site mask angle measured in mils, and it must be right justified in the field.

d. AD Radar. Hardware characteristics of the radars are entered in the Intelligence and Control Model constant data files (see Section IV, Chapter 6, Appendix A). These hardware characteristics are not to be exceeded by those in the Operating Instructions input. The operating instructions that the gamer is expected to provide on AD sensors are those that deal with operating constraints and changes in the situation as the battle progresses. This card format must be prepared for each AD radar to be played. Format of the input card is shown on Figure III-3-A-3.

(1) Sensor Code (Columns 1-2). Each generic type of sensor is identified by a sensor code. For AD radar, enter the value 08.



(2) UID of First Node in Communication Net (Columns 4-11). Enter the eight-character UID of the unit that will be the first node in a communication network from this specific radar site. This must be a maneuver unit (i.e., have the character M in the third character of the UTD).

(3) Unique Sensor Index (Columns 13-14). When this receiving node has more than one AD radar reporting to it, this index is used to discriminate among the radars. The index must be assigned serially, starting with 1 for the first AD radar reporting to this unit, 2 for the second, etc. In relocating or reorienting a radar, this index is used to decide which radar to move or reorient, should the unit have more than one reporting to it. The entry must be right justified.

(4) Sensor Reference Number (Columns 16-18). Each specific type of sensor hardware is parametrically defined within the constant data input on the Intelligence and Control Model data file. In the process, a reference number is assigned by the data load program, and it is this number that is required here to tie this particular radar to a set of hardware data.

(5) Sensor Status Code (Column 20). The status of the radar at the beginning of the game or period is to be entered in this card column. Entering a 2 or 3 will preclude the radar from detecting and recognizing targets during the course of the period. Entering a 1 will permit full play of the radar during the course of the period.

(6) Sensor Coordinates and Height (Columns 22-44). Enter the seven-digit X, Y coordinates and height in meters of the sensor right adjusted in the appropriate columns. Leading zeros are not required.

(7) Assigned Area of Search (Columns 46-76). The area of search to be assigned to an AD radar is specified by the entries in columns 46-76. Columns 46-50 and 52-56 contain the minimum and maximum ranges, respectively, of the assigned search area, in meters. Columns 58-61 contain the orientation, or central azimuth, of the assigned search sector, measured in mils clockwise from grid north. Columns 63-66 contain the width of the assigned search sector, in mils. Columns 68-71 contain the vertical central azimuth of the search area, in mils. Columns 73-76 contain the width of the vertical search sector, in mils.

e. Unattended Ground Sensors. This section describes the card formats required to represent unattended ground sensor fields. Each UGS field requires three data cards. The first card contains data necessary to identify the UGS field, the second card specifies the composition of the UGS field, and the third card specifies the location of the field. The format of each of these cards is discussed below.

(1) UGS Identification (Figure III-3-A-4). This card specifies the sensor code, UID of the controlling unit, unique sensor index, sensor status, and the median time to track and report enemy units.



GCS IDENTIFICATION									
SENSOR CODE	UID CONTROLLING UNIT	BLANK	SENSOR INDEX	BLANK	STATUS CODE	BLANK	MEDIAN TIME (SECS) TO TRACK AND DETECT	BLANK	BLANK
1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9
10	10	10	10	10	10	10	10	10	10
11	11	11	11	11	11	11	11	11	11
12	12	12	12	12	12	12	12	12	12
13	13	13	13	13	13	13	13	13	13
14	14	14	14	14	14	14	14	14	14
15	15	15	15	15	15	15	15	15	15
16	16	16	16	16	16	16	16	16	16
17	17	17	17	17	17	17	17	17	17
18	18	18	18	18	18	18	18	18	18
19	19	19	19	19	19	19	19	19	19
20	20	20	20	20	20	20	20	20	20
21	21	21	21	21	21	21	21	21	21
22	22	22	22	22	22	22	22	22	22
23	23	23	23	23	23	23	23	23	23
24	24	24	24	24	24	24	24	24	24
25	25	25	25	25	25	25	25	25	25
26	26	26	26	26	26	26	26	26	26
27	27	27	27	27	27	27	27	27	27
28	28	28	28	28	28	28	28	28	28
29	29	29	29	29	29	29	29	29	29
30	30	30	30	30	30	30	30	30	30
31	31	31	31	31	31	31	31	31	31
32	32	32	32	32	32	32	32	32	32
33	33	33	33	33	33	33	33	33	33
34	34	34	34	34	34	34	34	34	34
35	35	35	35	35	35	35	35	35	35
36	36	36	36	36	36	36	36	36	36
37	37	37	37	37	37	37	37	37	37
38	38	38	38	38	38	38	38	38	38
39	39	39	39	39	39	39	39	39	39
40	40	40	40	40	40	40	40	40	40
41	41	41	41	41	41	41	41	41	41
42	42	42	42	42	42	42	42	42	42
43	43	43	43	43	43	43	43	43	43
44	44	44	44	44	44	44	44	44	44
45	45	45	45	45	45	45	45	45	45
46	46	46	46	46	46	46	46	46	46
47	47	47	47	47	47	47	47	47	47
48	48	48	48	48	48	48	48	48	48
49	49	49	49	49	49	49	49	49	49
50	50	50	50	50	50	50	50	50	50
51	51	51	51	51	51	51	51	51	51
52	52	52	52	52	52	52	52	52	52
53	53	53	53	53	53	53	53	53	53
54	54	54	54	54	54	54	54	54	54
55	55	55	55	55	55	55	55	55	55
56	56	56	56	56	56	56	56	56	56
57	57	57	57	57	57	57	57	57	57
58	58	58	58	58	58	58	58	58	58
59	59	59	59	59	59	59	59	59	59
60	60	60	60	60	60	60	60	60	60
61	61	61	61	61	61	61	61	61	61
62	62	62	62	62	62	62	62	62	62
63	63	63	63	63	63	63	63	63	63
64	64	64	64	64	64	64	64	64	64
65	65	65	65	65	65	65	65	65	65
66	66	66	66	66	66	66	66	66	66
67	67	67	67	67	67	67	67	67	67
68	68	68	68	68	68	68	68	68	68
69	69	69	69	69	69	69	69	69	69
70	70	70	70	70	70	70	70	70	70
71	71	71	71	71	71	71	71	71	71
72	72	72	72	72	72	72	72	72	72
73	73	73	73	73	73	73	73	73	73
74	74	74	74	74	74	74	74	74	74
75	75	75	75	75	75	75	75	75	75
76	76	76	76	76	76	76	76	76	76
77	77	77	77	77	77	77	77	77	77
78	78	78	78	78	78	78	78	78	78
79	79	79	79	79	79	79	79	79	79
80	80	80	80	80	80	80	80	80	80
81	81	81	81	81	81	81	81	81	81
82	82	82	82	82	82	82	82	82	82
83	83	83	83	83	83	83	83	83	83
84	84	84	84	84	84	84	84	84	84
85	85	85	85	85	85	85	85	85	85
86	86	86	86	86	86	86	86	86	86
87	87	87	87	87	87	87	87	87	87
88	88	88	88	88	88	88	88	88	88
89	89	89	89	89	89	89	89	89	89
90	90	90	90	90	90	90	90	90	90
91	91	91	91	91	91	91	91	91	91
92	92	92	92	92	92	92	92	92	92
93	93	93	93	93	93	93	93	93	93
94	94	94	94	94	94	94	94	94	94
95	95	95	95	95	95	95	95	95	95
96	96	96	96	96	96	96	96	96	96
97	97	97	97	97	97	97	97	97	97
98	98	98	98	98	98	98	98	98	98
99	99	99	99	99	99	99	99	99	99
100	100	100	100	100	100	100	100	100	100

Figure III-3-A-4. Unattended Ground Sensor Identification Card Format

(a) Sensor Code (Columns 1-2). Each generic type of sensor is identified by a sensor code. For UGS fields, enter the value 07.

(b) UID of First Node in Communication Net (Columns 4-11). Enter the eight-character UID of the unit that will be the first node in a communication network from this specific UGS field. This unit must be a maneuver unit (i.e., have the character M in the third character of the UTD).

(c) Unique Sensor Index (Columns 13-14). When this receiving node has more than one UGS field reporting to it, this index is used to discriminate among the fields. The index must be assigned serially, starting with 1 for the first UGS field reporting to this unit, 2 for the second, etc.

(d) Sensor Status Code (Column 20). The status of the UGS field at the beginning of the game or period is to be entered in this card column. Entering a 2 or 3 will preclude the field from detecting and recognizing targets during the course of the period. Entering a 1 will permit full play of the field during the course of the period.

(e) Median Time to Track and Detect (Columns 39-44). Enter the median time in seconds, from activation of this UGS field to receipt of a sensing report at the first communication node.

(2) UGS Field Composition (Figure III-3-A-5). This card specifies the composition of the UGS field. Field composition is defined in terms of the number of each distinct type of UGS that is located within the field. Up to 10 different types of UGSs may be present in a given field. See Section IV. Appendix A to Chapter 6, for a discussion of UGS types.

(a) Sensor Type (Columns 2-3). The sensor type identifies the specific kind of UGS. This sensor type must agree with the definition given in the UGS constant data (Section 11, Chapter 6).

(b) Number (Columns 5-7). Enter the number of sensors, of the type specified above, to be placed in the UGS field.

(c) The remainder of the card has space for defining nine additional sensor types and their numbers.

(3) UGS Field Location (Figure III-3-A-6). This card specifies the battlefield locations of the UGS field by listing the four corner coordinates of the field. The shape of an UGS field is restricted to a convex quadrilateral. A geometric figure is said to be convex if it is possible to connect any two points in the figure by a straight line that lies entirely within the figure. The geometric figure shown as (a) below (Figure III-3-A-7) is convex; the figure shown as (b) is not. The order in which the corners are input is arbitrary.





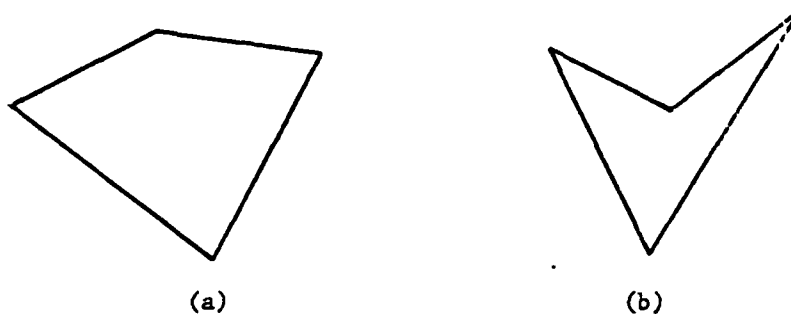


Figure III-3-A-7. Geometric Figures

(a) X coordinate (Columns 2-8). Enter the seven-digit X coordinate of the first corner of the field. Leading zeros are not required.

(b) Y coordinate (Columns 10-16). Enter the seven-digit Y coordinate of the first corner of the field.

(c) The remainder of the card contains the coordinates of the remaining three corners.

2. INTRODUCTION OF INTELLIGENCE. While the DIVWAG Period Processor simulates a range of ground-based and aerial sensors generally available to a division in the field, explicit simulation of all potential sources of battlefield intelligence is not attempted. The ability is available through the OPERINS loader to introduce additional intelligence to the model. This capability is intended to permit the control team to enrich the intelligence picture available to the opposing player teams in a closed or semi-closed game while maintaining the information within the format that is generally used for intelligence in the DIVWAG system. It also permits specification of targets for attack within the Air Ground Engagement Model or for automatic allocation of fire support, should this prove necessary. In all cases, use of this feature should be stringently controlled by the game's control team.

a. Approach. Intelligence introduced to the DIVWAG Period Processor through the OPERINS loader enters and is processed by the Intelligence and Control Model similarly to intelligence gathered by sources explicitly modeled. The data input by the OPERINS loader is formatted to produce a sensing report similar to those generated within the model. The input data format permits the user to control the time at which this sensing report enters the Intelligence and Control Model, the node in the intelligence communication network at which the sensing report enters the system, and whether the report passes through the intelligence processing portions of the model, the fire support allocation portions of the model, or both. Details of the Intelligence and Control Model communications network, processing, and fire support allocation logic are found in Section IV, Chapter 6.

b. Card Formats. Three card formats are prescribed for the introduction of intelligence with the OPERINS loader. The period card is prepared only once for each game period in which intelligence is introduced. For each sensing report to be introduced, one intelligence information card and one estimated equipment card are produced.

(1) Period Card. The period card, illustrated in Figure III-3-A-8, is the first data card of an intelligence data deck processed by the OPERINS loader. The information provided on this card is needed to correctly schedule time of entry of sensing reports into the Intelligence and Control Model. The symbols PERIOD, DAY, HOUR, MINUTE and a period are preprinted on the coding form at columns 1-6, 12-14, 19-22, 27-32, and 33 respectively and must be punched on the data card. Data to be provided are the starting time of the game period for which intelligence is to be introduced. Start of period in terms of day, hour, and minute are placed in columns 9-10, 16-17, and 24-25, respectively. Each entry is right justified.

(2) Intelligence Information. This card format, shown in Figure III-3-A-9, is the first of two cards required for each item of intelligence to be introduced into the system.

(a) Source Type Code (Columns 1-2). The source type code is a gamer or control group classification of the information source. Within the model, its function is simply one of mechanical bookkeeping; and processing of each sensing report is identical, regardless of the nature of the information source specified. Legal codes and their nominal source interpretation are:

- 31 = Prisoners of war
- 32 = Civilians
- 33 = Recovered military personnel
- 34 = Captured enemy documents
- 35 = Enemy materiel
- 36 = Stay-behind units
- 37 = Agencies for operation behind enemy lines
- 38 = Other sources.

(b) Where to Code (Columns 4-7). This code establishes the type of action to be taken upon the sensing report within the Intelligence and Control Model. Legal codes and indicated actions are:

- . INTL - Enter report only in intelligence processing and communication net
- . FSCC - Enter report only in the fire support allocation logic of the model
- . BOTH - Enter report both in intelligence processing and communication net and in fire support allocation logic.







(c) Source UID (Columns 9-16). Enter the eight-character alphanumeric UID of a unit to whom the information is first available. This unit will function as the first node in the communication network for the sensing report and must be a battalion, brigade, or division-level maneuver unit.

(d) Target UID (Columns 18-25). Enter the eight-character alphanumeric UID of the target unit; i.e., the unit described by this report.

(e) Time (Columns 27-33). Enter the time at which the information is to enter the Intelligence and Control Model. Upon entry the report is processed as any sensing report in the model, and the user must be aware of decision and processing delay times that will be applied. Enter the day in columns 27-28, the slash (/) in column 29, and the time (24-hour clock) in columns 30-33. For example, 02/0830 represents 0830 hours on the second day of simulated combat.

(f) Estimated Location (Columns 35-49). Enter the seven-digit X and Y coordinates of the estimated location of the target unit. Entries are right justified in columns 35-41 (X coordinate) and 43-49 (Y coordinate) and leading zeroes may be omitted.

(g) Estimated Activity (Columns 51-54). The estimated activity of the target unit at time of entry of the report into the system is required. Legal activity codes for the unit are:

MOVE	=	Unit moving
FIRT	=	Unit firing tube artillery
FIRM	=	Unit firing missile artillery
FIRA	=	Unit firing air defense weapons
ATCK	=	Unit in an attack posture
DEFN	=	Unit in a defensive posture
WTHD	=	Unit withdrawing
ENGR	=	Unit involved in engineering activity
STAY	=	Unit stationary and not otherwise active.

(h) Estimated Move Rate (Columns 56-57). If the estimated activity is MOVE, ATCK, or WTHD, enter the estimated rate of unit movement in meters per minute; otherwise, leave the field blank.

(i) Estimated Direction (Columns 61-63). If the estimated activity is MOVE, ATCK, or WTHD, enter the estimated direction of movement; otherwise, leave the field blank. Directions are expressed as points of the sixteen-point compass, and entries are right justified within the field. Legal entries are:

N	E	S	W
NNE	ESE	SSW	WNW
NE	SE	SW	NW
ENE	SSE	WSW	NNW

(3) Estimated Equipment. This card format, illustrated in Figure III-3-A-10, is the second of two cards required for each item of intelligence to be introduced into the system. This information is used within the model to develop estimated type and size of the target unit.

(a) Personnel (Columns 1-4). Estimated number of personnel obtained from source.

(b) Vehicles (Columns 6-8). Estimated number of vehicles, other than those in categories described below, obtained from source.

(c) Tanks (Columns 10-12). Number of tanks estimated by source about the target.

(d) APCs (Columns 14-16). Estimated number of APCs in target obtained from source.

(e) Artillery Tubes (Columns 18-20). Number of artillery tubes estimated by information source.

(f) Artillery Missiles (Columns 22-24). Estimated number of artillery missiles in target.

(g) Air Defense Guns (Columns 26-28). Number of AD guns estimated by source to be in target unit.

(h) Air Defense Missiles (Columns 30-32). Estimated number of AD missiles in target unit.

(i) Aircraft (Columns 34-36). Estimated number of aircraft obtained from information source.

### 3. FIRE SUPPORT ALLOCATION CONTROL:

a. Helicopter and Artillery Penetration Limits. This card format, shown in Figure III-3-A-11, sets limiting ranges beyond which helicopter and artillery fire support missions will not be requested by the fire support allocation logic of the Intelligence and Control Model. Ranges are stated in terms of distance beyond the FEBA. The Period Processor actually calculates a rectangular approximation of the trace of the FEBA to be used in reaching fire support allocation decisions. Since the model's approximation of the FEBA may not coincide with the FEBA on game maps plotted by the user, the calculation of the model FEBA, as described in Chapter 4 of Section IV, Battlefield and Unit Geometry, should be understood prior to assignment of these ranges. Ranges are initially set to zero within the model and should be set to desired values at the first game period.

(1) Blue Helicopters (Columns 2-7). Enter in meters the maximum range beyond the FEBA to which Blue attack helicopter missions may be automatically requested within the model.



HELICOPTER AND ARTILLERY PENETRATION LIMITS (METERS)									
BLUE					RED				
PENETRATION INTO ENEMY TERRITORY					PENETRATION INTO BLUE TERRITORY				
FOR HELICOPTERS	FOR ARTY TO FIRE	FOR HELICOPTERS	FOR ARTY TO FIRE	FOR HELICOPTERS	FOR HELICOPTERS	FOR ARTY TO FIRE	FOR HELICOPTERS	FOR ARTY TO FIRE	FOR HELICOPTERS
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100
101	102	103	104	105	106	107	108	109	110
111	112	113	114	115	116	117	118	119	120
121	122	123	124	125	126	127	128	129	130
131	132	133	134	135	136	137	138	139	140
141	142	143	144	145	146	147	148	149	150
151	152	153	154	155	156	157	158	159	160
161	162	163	164	165	166	167	168	169	170
171	172	173	174	175	176	177	178	179	180
181	182	183	184	185	186	187	188	189	190
191	192	193	194	195	196	197	198	199	200
201	202	203	204	205	206	207	208	209	210
211	212	213	214	215	216	217	218	219	220
221	222	223	224	225	226	227	228	229	230
231	232	233	234	235	236	237	238	239	240
241	242	243	244	245	246	247	248	249	250
251	252	253	254	255	256	257	258	259	260
261	262	263	264	265	266	267	268	269	270
271	272	273	274	275	276	277	278	279	280
281	282	283	284	285	286	287	288	289	290
291	292	293	294	295	296	297	298	299	300
301	302	303	304	305	306	307	308	309	310
311	312	313	314	315	316	317	318	319	320
321	322	323	324	325	326	327	328	329	330
331	332	333	334	335	336	337	338	339	340
341	342	343	344	345	346	347	348	349	350
351	352	353	354	355	356	357	358	359	360
361	362	363	364	365	366	367	368	369	370
371	372	373	374	375	376	377	378	379	380
381	382	383	384	385	386	387	388	389	390
391	392	393	394	395	396	397	398	399	400
401	402	403	404	405	406	407	408	409	410
411	412	413	414	415	416	417	418	419	420
421	422	423	424	425	426	427	428	429	430
431	432	433	434	435	436	437	438	439	440
441	442	443	444	445	446	447	448	449	450
451	452	453	454	455	456	457	458	459	460
461	462	463	464	465	466	467	468	469	470
471	472	473	474	475	476	477	478	479	480
481	482	483	484	485	486	487	488	489	490
491	492	493	494	495	496	497	498	499	500
501	502	503	504	505	506	507	508	509	510
511	512	513	514	515	516	517	518	519	520
521	522	523	524	525	526	527	528	529	530
531	532	533	534	535	536	537	538	539	540
541	542	543	544	545	546	547	548	549	550
551	552	553	554	555	556	557	558	559	560
561	562	563	564	565	566	567	568	569	570
571	572	573	574	575	576	577	578	579	580
581	582	583	584	585	586	587	588	589	590
591	592	593	594	595	596	597	598	599	600
601	602	603	604	605	606	607	608	609	610
611	612	613	614	615	616	617	618	619	620
621	622	623	624	625	626	627	628	629	630
631	632	633	634	635	636	637	638	639	640
641	642	643	644	645	646	647	648	649	650
651	652	653	654	655	656	657	658	659	660
661	662	663	664	665	666	667	668	669	670
671	672	673	674	675	676	677	678	679	680
681	682	683	684	685	686	687	688	689	690
691	692	693	694	695	696	697	698	699	700
701	702	703	704	705	706	707	708	709	710
711	712	713	714	715	716	717	718	719	720
721	722	723	724	725	726	727	728	729	730
731	732	733	734	735	736	737	738	739	740
741	742	743	744	745	746	747	748	749	750
751	752	753	754	755	756	757	758	759	760
761	762	763	764	765	766	767	768	769	770
771	772	773	774	775	776	777	778	779	780
781	782	783	784	785	786	787	788	789	790
791	792	793	794	795	796	797	798	799	800
801	802	803	804	805	806	807	808	809	810
811	812	813	814	815	816	817	818	819	820
821	822	823	824	825	826	827	828	829	830
831	832	833	834	835	836	837	838	839	840
841	842	843	844	845	846	847	848	849	850
851	852	853	854	855	856	857	858	859	860
861	862	863	864	865	866	867	868	869	870
871	872	873	874	875	876	877	878	879	880
881	882	883	884	885	886	887	888	889	890
891	892	893	894	895	896	897	898	899	900
901	902	903	904	905	906	907	908	909	910
911	912	913	914	915	916	917	918	919	920
921	922	923	924	925	926	927	928	929	930
931	932	933	934	935	936	937	938	939	940
941	942	943	944	945	946	947	948	949	950
951	952	953	954	955	956	957	958	959	960
961	962	963	964	965	966	967	968	969	970
971	972	973	974	975	976	977	978	979	980
981	982	983	984	985	986	987	988	989	990
991	992	993	994	995	996	997	998	999	1000

Figure III-3-A-11. Helicopter and Artillery Penetration Limits Card Format

(2) Blue Artillery (Columns 9-14). Enter in meters the maximum range beyond the FEBA to which Blue artillery fire missions may be automatically requested within the model.

(3) Red Helicopters (Columns 16-21) and Red Artillery (Columns 23-28). Same definition as columns 2-7 and 9-14 above.

b. Close Air Support Sorties. These card formats are used to set the number of TACAIR sorties available to the Blue and Red forces. They are initialized at zero and must be set by OPERINS data to obtain TACAIR simulation.

(1) Blue Sortie Constraints (Figure III-3-A-12). It is assumed that all Blue sorties will be conducted by all-weather-capable aircraft. Sorties are allocated in 6-hour blocks, starting at midnight; and unused sorties in one block are not carried over to later blocks. Entries are the maximum number of sorties available from 0001 to 0600 (columns 1-5); from 0601 to 1200 (columns 6-10); from 1201 to 1800 (columns 11-15); and from 1801 to 2400 (columns 16-20).

(2) Red Sortie Constraints (Figure III-3-A-13). Both all-weather-capable aircraft and aircraft limited to good visibility conditions are played for Red. The good visibility aircraft are flown first, conditions and sortie limits permitting; otherwise, allocation of sorties is as for Blue TACAIR. Enter maximum Red all-weather-capable sorties for the time blocks 0001 to 0600, 0601 to 1200, 1201 to 1800, and 1801 to 2400 in columns 1-5, 6-10, 11-15, and 16-20 respectively. Enter maximum Red sorties by aircraft limited to good visibility for the time blocks 0001 to 0600, 0601 to 1200, 1201 to 1800, and 1801 to 2400 in columns 21-25, 26-30, 31-35, and 36-40 respectively.

4. OPERINS DATA DECK STRUCTURE. The data deck for the OPERINS loader is composed of up to four subdecks. Any or all of these decks can be provided for one operation of the loader. Each subdeck is introduced by a header card with the proper code punched in the first two columns. Legal header card codes and associated subdecks are:

- 01 = Sensor control subdeck
- 02 = Introduction of intelligence subdeck
- 03 = Helicopter and artillery penetration subdeck
- 04 = Close air support sorties subdeck.

a. Subdeck Structure:

(1) Sensor Control Subdeck. The first card of the sensor control subdeck is a header card with the symbols 01 punched in the first two card columns. Following the header card, sensor control data cards may be provided in any sequence with the requirement that the three cards required to establish an UGS field must be contiguous and in the order described in subparagraph 1e (UGS Identification, UGS Field Composition, UGS Field Location) for each UGS field. The user should bear in mind the limit of 100 model





sensors (radars or UGS fields) per force. The size of this subdeck is not fixed; and the last card of the subdeck must be a blank card, used to signal the end of the subdeck to the loader.

(2) Introduction of Intelligence Subdeck. The first card of this subdeck must be the header card with the symbols 02 punched in the first two card columns. The second card of this subdeck must be the period card, described in subparagraph 2b(1). For each sensing report to be introduced into the system, the intelligence information card must be followed immediately by the estimated equipment card. There is no practical limit on the number of such card pairs to be processed. The size of this subdeck is not fixed; and the last card of the subdeck must be a blank card, used to signal the end of the subdeck to the loader.

(3) Helicopter and Artillery Penetration Subdeck. This subdeck consists of the header card, with 03 punched in the first two card columns, followed by one data card as discussed in subparagraph 3a.

(4) Close Air Support Sorties Subdeck. This subdeck is composed of a header card, with 04 punched in the first two card columns, followed by a Blue sortie card and a Red sortie card. All three cards must be present in the proper order.

b. Deck Structure. Only those subdecks required for the game period need be included in the data deck. The order of subdecks within the data deck is of no significance as long as each subdeck starts with the proper header card. An end of data deck card, with 99 punched in columns 1-2 of the card, must follow the last subdeck. A typical data deck for Operating Instructions is shown in Figure III-3-A-14.



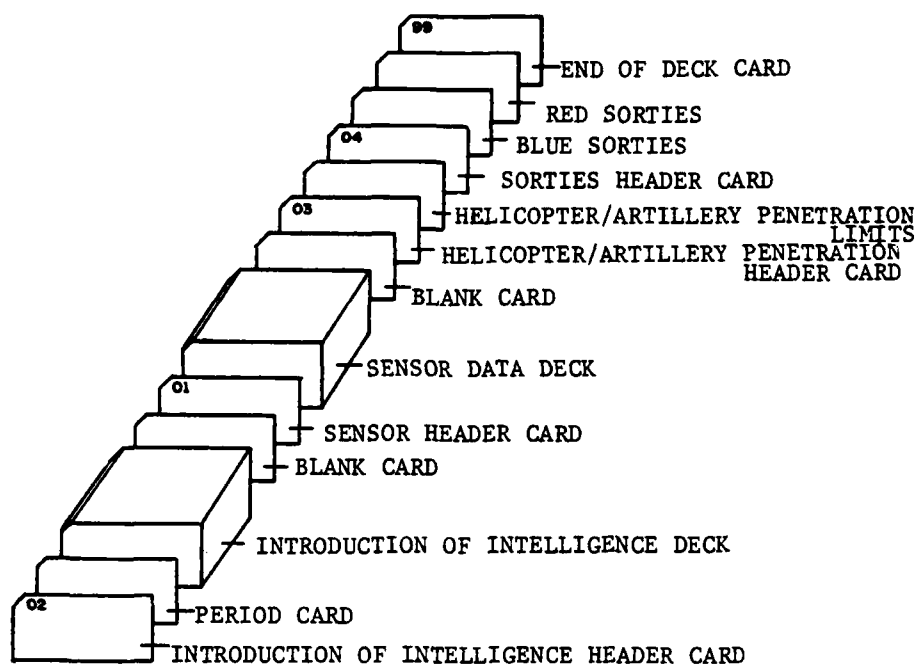


Figure III-3-A-14. Data Deck Arrangement for Operating Instructions

NOTES

III-3-A-26

## APPENDIX B

### OPERATING INSTRUCTIONS PROGRAM DESCRIPTIONS

1. INTRODUCTION. The routines described in this appendix make up the operating instructions loader and include a loader executive, DATAST, and a group of routines that read data cards and place the data directly into the DIVWAG data files. These routines are considered part of the Orders Input Processor because they may be used at the beginning of any game period.

#### 2. ROUTINE DATAST:

a. Purpose. DATAST is the executive routine of the operating instructions loader.

##### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IFNT(56,3)	Disk	File name table.
UTDTAB(1000)	DF36	Unit type designator (UTD) for all units.
UIDTAB(2,1000)	DF36	Unit identification cross-reference table for all units.
IBLUE	DF36	IUID of last Blue unit.
IRED	DF36	IUID of first Red unit.
JHEAD	Card	Header card to indicate the type of load required.

##### c. Output Variables. None.

##### d. Logical Flow (Figure III-3-B-1):

(1) Block 1. Call routine GETFLE to load the file name table.

(2) Block 2. Call routine GETWRD to obtain from data file 36 the unit type designator cross-reference table, the unit identification cross-reference table, and the last used IUID for both Blue and Red forces.

(3) Block 4. Zero out the array of sequence numbers for sensing reports from all battlefield sources and call routine PUTRCD to place this array in data file 16.

(4) Block L1000. Read header card to indicate which set of data is to follow.

(5) Block 5. If the header card is an end-of-data card, terminate the program.

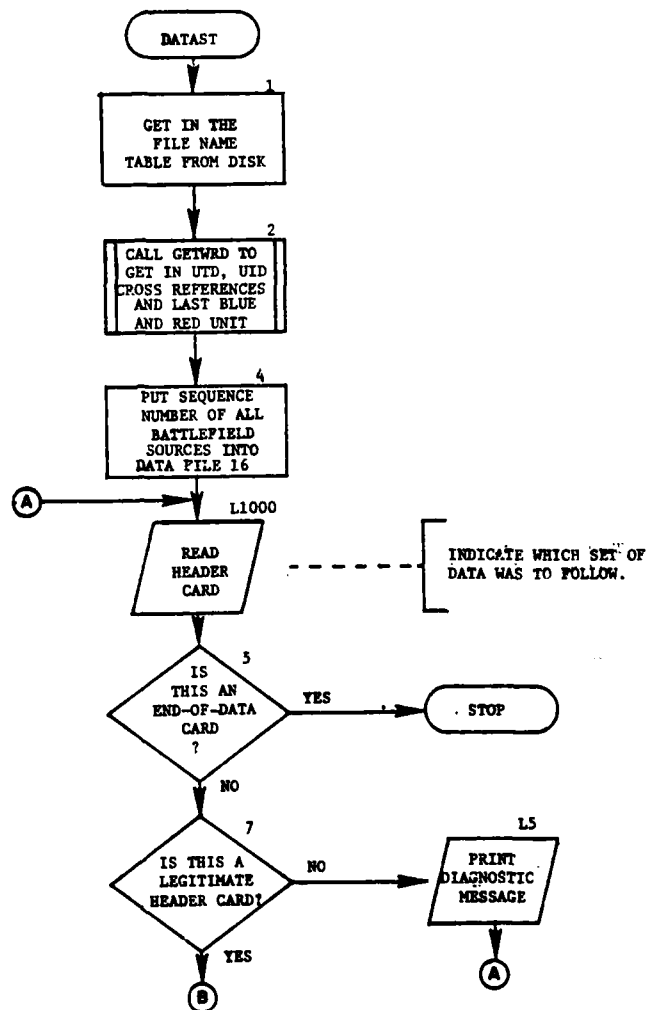


Figure III-3-B-1. Routine DATAS (Continued on Next Page)

AD-A112 686

ARMY COMBINED ARMS COMBAT DEVELOPMENTS ACTIVITY FORT--ETC F/8 1B/7  
DIWAS MODEL DOCUMENTATION. VOLUME II. PROGRAMMER/ANALYST MANUAL--ETC(U)  
JAL 76

UNCLASSIFIED

CACDA-TR-8-76-VOL-2-PT-1

NL

9  
21 24 25A

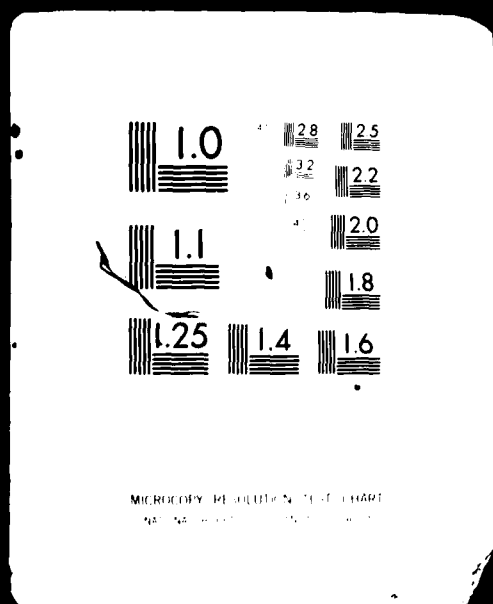
END  
DATE  
FILMED  
82  
DTIC

# SS FILE

# gOLF

AD

A/12658



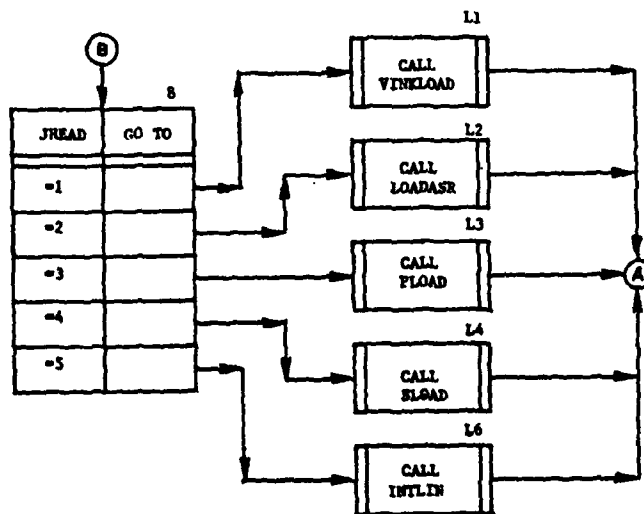


Figure III-3-B-1. Routine DATAST (Concluded)

(6) Blocks 7 and L5. If the header card is a legitimate header card (i.e., contains a value between one and five inclusive), control goes to block 8; otherwise, print a diagnostic message about the illegal header card and transfer control to block L1000 to read the next card.

(7) Block 8. Control branches according to value in header card. If JHEAD is equal to one, control goes to block L1. If JHEAD is equal to 2, control goes to block L6. If JHEAD is equal to three, control goes to block L3. If JHEAD is equal to four, control goes to block L4. If JHEAD is equal to five, control goes to block L2.

(8) Block L1. Call VINKLD to load data file 20 with the individual sensor data. Upon its return, control goes to block L1000.

(9) Block L2. Call LDASR to load a unit's available supply rates (ASR). Upon controls return, it goes to block L1000.

(10) Block L3. Call PLOAD to load the ranges from the FEBA for the decision process of the Intelligence and Control Model on the type of fire support to request. Upon its return, control goes to block L1000.

(11) Block L4. Call SLOAD to load the Blue and Red sorties for a period of time in 6-hour blocks. Upon its return, control goes to block L1000.

(12) Block L6. Call INTLIN to load intelligence information. Upon its return, control goes to block L1000.

### 3. ROUTINE VINKLD:

a. Purpose. VINKLD loads the individual sensor data desired for this game period onto data file 20.

#### b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
IDUM(320)	DF20	Constant unattended ground sensor (UGS) field data. First 160 words describe Blue UGS fields and last 160 words describe Red UGS fields.
INVEC(15)	Card	Sensor data for individual sensor read from cards.
UTDTAB(1000)	TWO	Unit type designator table for all units.
JREC(20)	DF20	Individual sensor data record from data file 20.

#### c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
JREC(20)	DF20	New or updated individual sensor data record.



d. Logical Flow (Figure III-3-B-2):

- (1) Block 1. Get from data file 20 the constant unattended ground sensor field data (loaded by INCLD).
- (2) Block L1000. Read sensor data card.
- (3) Block 2. If all the data have been read (sensed by blank card), control goes to block L6000.
- (4) Block 3. Print the contents of the data card just read.
- (5) Block 4. Utilizing the unit identification read as the first node in the communication network, determine the IUID and, using it, obtain the unit type designator of the first node from the unit type designator table.
- (6) Block 5. If the first node is not a maneuver unit (last characters of unit type designator not MI, MT, or MC), control goes to block L5000.
- (7) Block 6. Set the sensor record pointer for the Blue or Red force.
- (8) Block 7. Loop through the individual sensor records to determine if this is a new set of data or an update to an existing record.
- (9) Blocks 8 and 9. If this is a new set of data, transfer control to block L2. If this is an update set of data, transfer control to block L3.
- (10) Block 10. Since the data are not an update to existing sensor data and there is no room on the existing sensor data record for new sensor data, a diagnostic message is written and control is returned to the calling routine.
- (11) Block L2. Set the new data flag and load the keyword into the sensor record array.
- (12) Block L3. Compute the next record counter.
- (13) Block 11. If the new sensor data are not for an update, transfer control to block L7.
- (14) Block 12. If the new set of data is for an unattended ground sensor field sensor, transfer control to block L300.
- (15) Block L10. Set up sensor record with parameters that are common to the moving target indicator, air defense, and countermortar or counterbattery radars.
- (16) Block 13. If this is a countermortar or counterbattery radar type sensor, transfer control to block L90.
- (17) Block 14. Set those sensor values common to the moving target indicator and air defense radars.

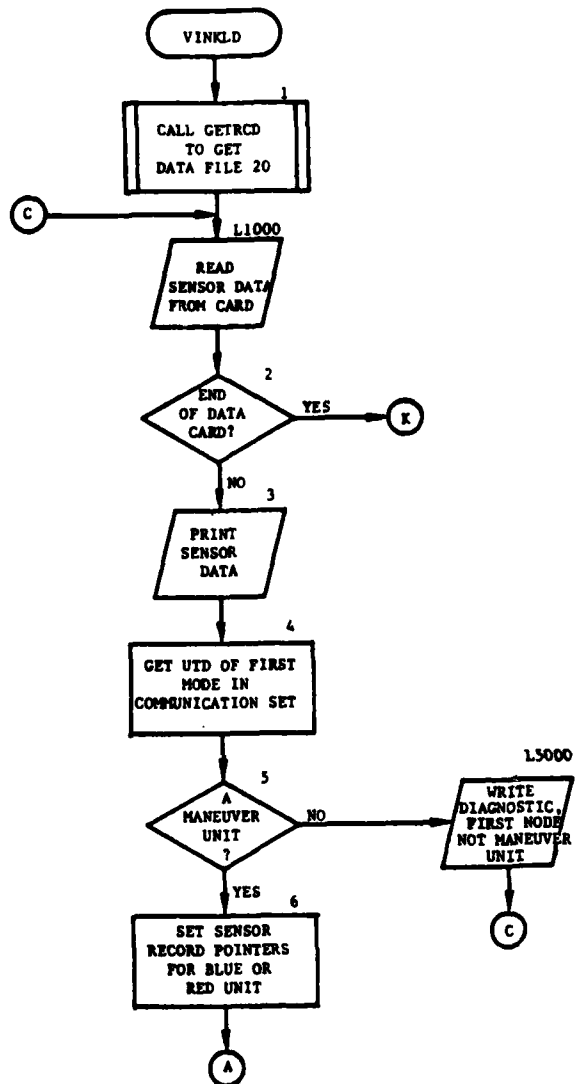


Figure III-3-B-2. Routine VINKLD (Continued on Next Page)

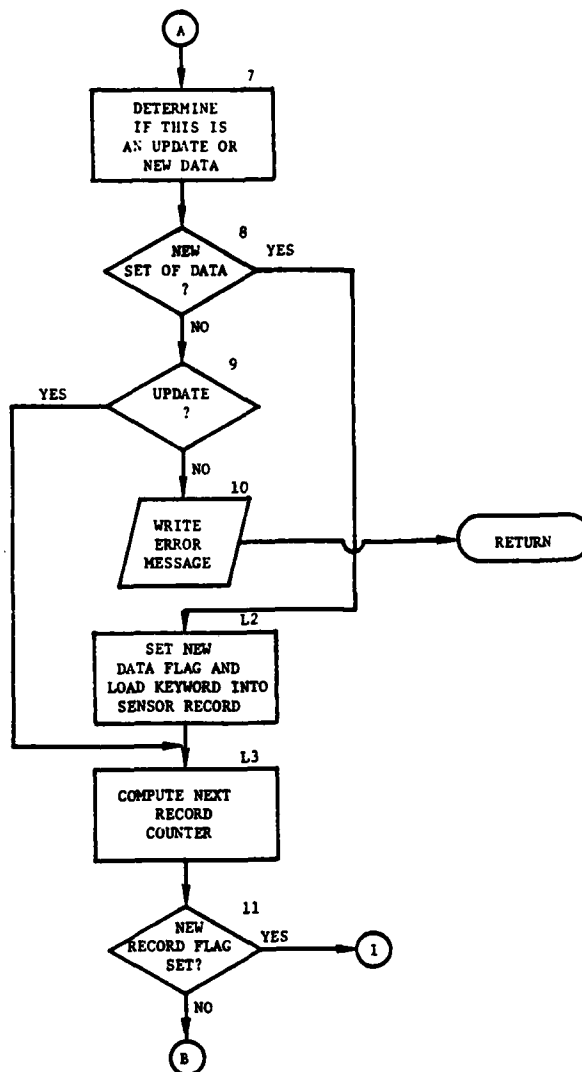


Figure III-3-B-2. Routine VINKLD (Continued)

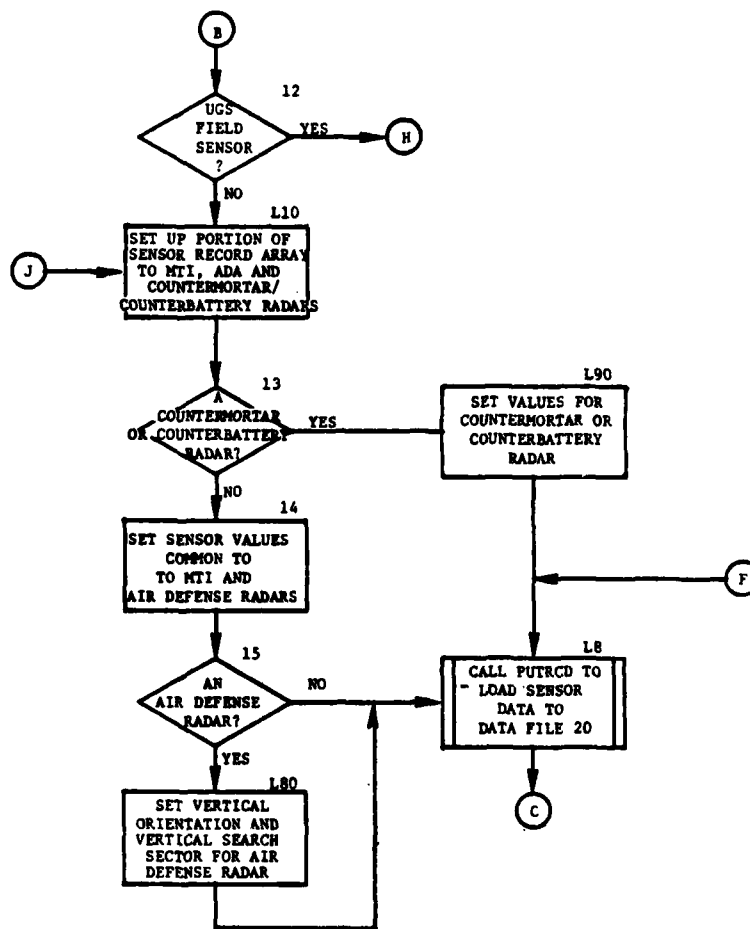


Figure III-3-B-2. Routine VINKLD (Continued)

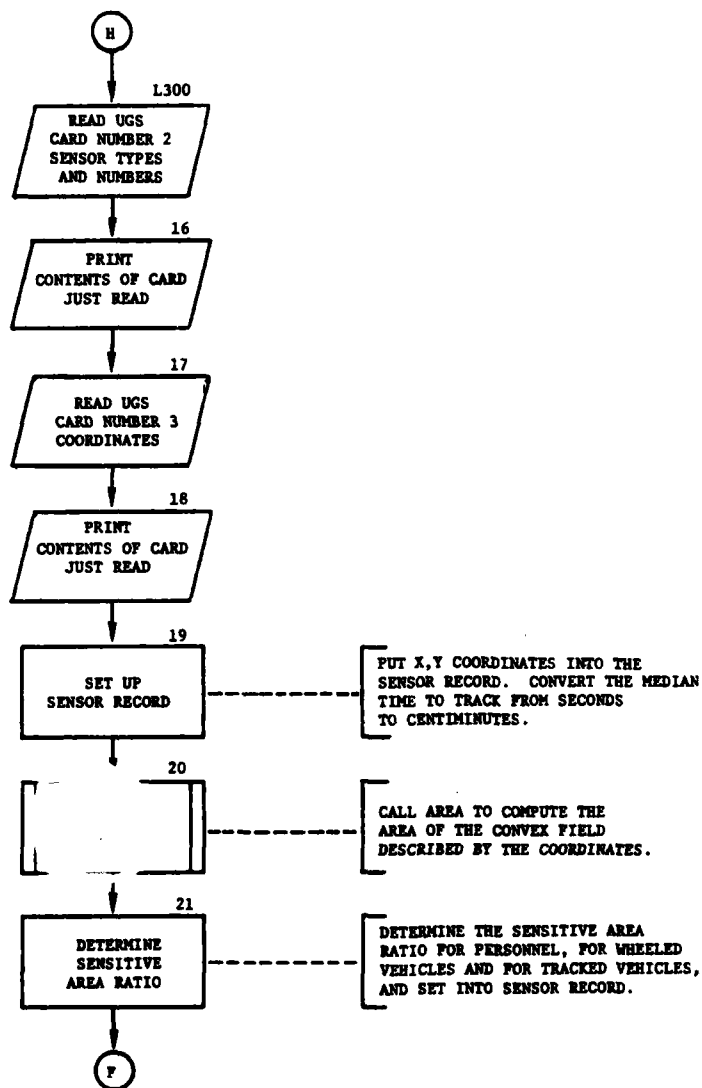


Figure III-3-B-2. Routine VINKLD (Continued)

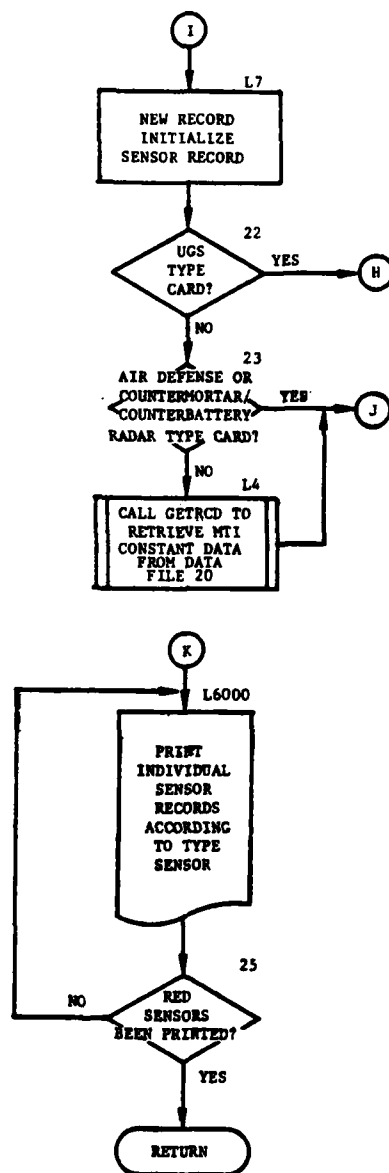


Figure III-3-B-2. Routine VINKLD (Concluded)

(18) Block L5. If this is not an air defense radar sensor type, transfer control to block L8.

(19) Block L80. Set the vertical orientation and vertical sector for air defense radar.

(20) Block L8. Put individual sensor record onto data file 20 and return control to block L1000.

(21) Block L90. Set the values of the countermortar/counterbattery radar, and transfer control to block L8.

(22) Blocks L300, 16, 17, and 18. Read cards 2 and 3 for unattended ground sensor fields and print their contents. Card 2 contains sensor types and numbers to make up an unattended ground sensor field. Card 3 contains the four X,Y coordinates that give the unattended ground sensor field its convex geometric shape.

(23) Block 19. Put the coordinates into the sensor record. Convert the median time to track from seconds to centiminutes and store into sensor record.

(24) Blocks 20 and 21. Call routine AREA to determine the area of the unattended ground sensor field described by the coordinates and determine the sensitive area ratios for personnel, for wheeled vehicles, and for tracked vehicles; place these into the sensor record, and transfer control to block L8.

(25) Block L7. Zero sensor record, and set up values common to all sensor types.

(26) Blocks 22 and 23. If this is an unattended ground sensor radar card type, transfer control to block L300. If this is an air defense or countermortar/counterbattery radar type card, transfer control to block L10.

(27) Block 24. Call routine GETRCD to retrieve the minimum radial velocity, azimuth auto scan rate, and range auto scan depth; then, transfer control to block L10.

(28) Block L5000. Print a diagnostic message that the first node in the communication network is not a maneuver unit; then transfer control to block L1000 to read the next card.

(29) Blocks L6000 and 25. Print each sensor record formatted according to sensor type; first, the Blue force sensors and then the Red force sensors are printed. Return control to the calling routine.

#### 4. ROUTINE CNVRT:

a. Purpose. CNVRT is called by VINKLD to convert angles in mils measured clockwise from grid north to angles in radians measured counterclockwise from the positive X axis.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ANG	Call	Angle in mils measured clockwise from grid north.

c. Output Variable:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
CNVRT	Call	Angle in radians measured counterclockwise from the positive X axis.

d. Processing Description. The angle is converted from mils to degrees from grid north, using the multiplicative factor 0.05625. This is converted to degrees from the X axis by a change in sign and addition of 45° (90° if the original angle was less than or equal to 90°). Finally, the result is converted to radians by the multiplicative factor 0.01745.

5. ROUTINE AREA:

a. Purpose. AREA is called by VINKLD to compute the area of a convex quadrilateral by summing the areas of the two triangles formed by bisecting the quadrilateral.

b. Input Variable:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
X(4), Y(4)	ONE	Set of four (X,Y) coordinate locations of the corners of a convex figure describing the unattended ground sensor field.

c. Output Variable:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
ANS	Call	Area of the convex quadrilateral computed by this routine.

d. Processing Description. The lengths of each side of the quadrilateral are determined as A, B, C, and D and the length of a diagonal as E, all using the Pythagorean theorem. (The quadrilateral is bisected into triangles with sides A, B, E, and C, D, E.) The area of each triangle is calculated, and the two areas are summed.

6. ROUTINE LDASR:

a. Purpose. LDASR loads from cards the available supply rate for specific equipment types for a specified unit.



b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
UID	Card	Unit identifier code for unit receiving the available supply rate.
NI	Card	The number of entries on the data card.
INDEX(100)	Card	An array of equipment item codes.
ASR(100)	Card	An array of associated available supply rates.
UMAIN(500)	DF1	Unit status file of unit receiving the available supply rates.
LIST(10)		UMAIN(277). List of subordinate units.
PNT1, PNT2	ONE	UMAIN(317) and UMAIN(318). Beginning and ending supply status records.
EQUIP(1000)	DF31	Storage for up to 100 ten-word records from data file 31.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
EQUIP(1000)	DF31	Updated available supply rate data in storage area for up to 100 ten-word records for data file 31.

d. Logical Flow (Figure III-3-B-3):

(1) Block L1000. Read the card containing the following information: UID, unit identification of unit to receive the new available supply rates; NI, number of new available supply rate entries; INDEX, an array of equipment item codes; and ASR, an array of associated available supply rates.

(2) Block 1. If this is an end-of-data card (i.e., blank card), then return control to the calling routine.

(3) Block 2. Print the contents of the card just read.

(4) Block L1001. Get from data file 1 the unit status record of the unit receiving the new available supply rates.

(5) Block 3. Utilizing the record pointers of this unit's data file 31 records, get this unit's supply status record from data file 31.

(6) Block 4. Search for a match between the item codes read from cards and those on the supply status array. When there is a match, the new available supply rate associated with the item code is stored into the proper location on the data file 31 storage array.

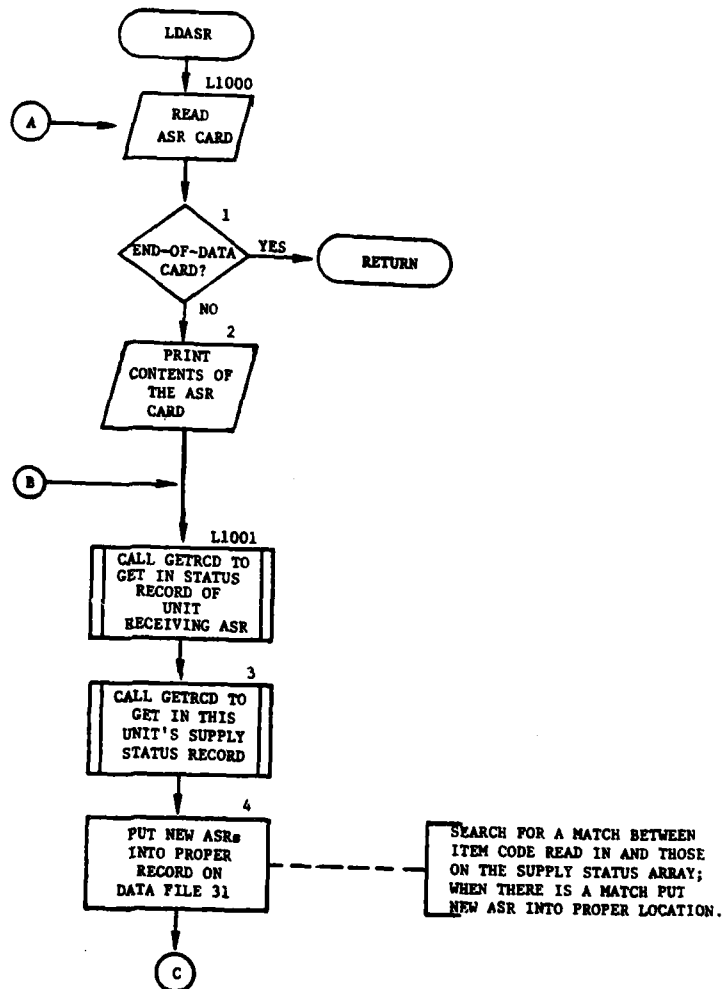


Figure III-3-B-3. Routine LDASR (Continued on Next Page)

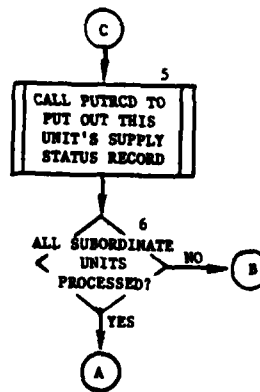


Figure III-3-B-3. Routine LDASR (Concluded)

(7) Block 5. Put this unit's updated supply status file on data file 31.

(8) Block 6. If this unit has subordinate units, then their supply status records must be updated to correspond with those of the superior; therefore, transfer control to block L1001. If all the subordinates have been processed or there are no subordinates, then transfer control to block L1000 for the next card.

7. ROUTINE PLOAD:

a. Purpose. PLOAD loads ranges from the forward edge of battle area used in the decision process as to what fire support to request.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
JBLU(2)	Card	Blue force attack helicopter range limit (JBLU(1)) and artillery range limit (JBLU(2)).
JRED(2)	Card	Red force attack helicopter range limit (JRED(1)) and artillery range limit (JRED(2)).

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
JBLUE(2)	DF16	Same as input variable description.
JRED(2)	DF16	Same as input variable description.

d. Processing Description. A data card is read, values on the card are listed and placed on data file 16, and control returns to the calling routine.

8. ROUTINE SLOAD:

a. Purpose. SLOAD reads the Blue and Red close air support (CAS) sortie constraints and stores them on data file 26.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
JSORTY(8)	Card	JSORTY (1 through 4) are good weather sorties for a 24-hour period, allocated in 6-hour segments. JSORTY (5 through 8) are all-weather sorties for a 24-hour period, allocated in four 6-hour blocks.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
JSORTY(8)	DF26	Same as described for input variables.

d. Processing Description. A card is read and listed, and its contents placed on data file 26 as Blue sortie limits. A second card is read and listed, and its contents placed on data file 26 as Red sortie limits. Control then returns to the calling routine.

9. ROUTINE INTLIN:

a. Purpose. The Intelligence and Control Model utilizes simulation of enemy activity detecting devices to develop an intelligence picture. Although these provide valuable combat intelligence information, they are not the only source of intelligence. INTLIN allows introduction of intelligence information from other battlefield sources.

b. Input Variables:

<u>Name</u>	<u>Source</u>	<u>Contents</u>
ISENST	Card	The source type code is the classification of the information source.
IWHERE	Card	Code indicating where the information is to go within the Intelligence and Control Model.
SUID	Card	Identification of the unit to which the information will first be communicated (i.e., first node in the information network; hence, must be a maneuver type unit).
TUID	Card	Identification of the unit that is described in the information (i.e., the target UID).
IDAY and IHRMIN	Card	Time at which the information enters the Intelligence and Control Model. It is also the time that will appear on the intelligence report as the detection time.
ESTX	Card	Estimated X coordinate of the target (detected) unit in meters, (up to seven digits).
ESTY	Card	Estimated Y coordinate of the target unit in meters.
IACT	Card	The estimated activity code is a four-character code describing the target unit's estimated activity.

<u>Name</u>	<u>Source</u>	<u>Contents</u>
MOVRAT	Card	Estimated movement rate in meters/minute, if the estimated activity is movement or if the target unit is attacking.
IDIR	Card	Estimated direction of the movement, if the target is considered to be moving.
NOPERS	Card	Estimated number of personnel in target.
NOVEH	Card	Estimated number of wheeled vehicles in target.
NOTNKS	Card	Estimated number of tanks in target.
NOAPCS	Card	Estimated number of armored personnel carriers in target.
NOATBS	Card	Estimated number of artillery tubes in target.
NOAMIS	Card	Estimated number of artillery missiles in target.
NOADGN	Card	Estimated number of air defense guns in target.
NOADMS	Card	Estimated number of air defense missiles in target.
NOACFT	Card	Estimated number of aircraft in target.

c. Output Variables:

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
FILE12(35)	DF12	Automatic event array containing intelligence information to be entered into the Intelligence and Control Model.
LUID		FILE12(1). Identification of the detecting unit.
INCOPR		FILE12(2). Intelligence and Control function to be performed.
RPTNO		FILE12(5). Sensing report number.
MUID		FILE12(6). Identification of the unit sensed.
ESTX		FILE12(7). Estimated X coordinate.
ESTY		FILE12(8). Estimated Y coordinate.
ESTSZE		FILE12(9). Estimated size of unit sensed.

<u>Name</u>	<u>Destination</u>	<u>Contents</u>
ESTYPE		FILE12(10). Estimated type of unit sensed.
ESTACT		FILE12(11). Estimated activity of unit sensed.
MOVRAT		FILE12(12). Average movement rate of detected unit.
ESTDIR		FILE12(13). Estimated direction of movement, if moving.
NOPERS		FILE12(14). Estimated number of personnel detected.
NOVEH		FILE12(15). Estimated number of vehicles detected.
NOTNKS		FILE12(16). Estimated number of tanks detected.
NOAPCS		FILE12(17). Estimated number of armored personnel carriers detected.
NOATBS		FILE12(18). Estimated number of artillery tubes detected.
NOAMIS		FILE12(19). Estimated number of artillery missiles detected.
NOADGN		FILE12(20). Estimated number of air defense guns detected.
NOADMS		FILE12(21). Estimated number of air defense missiles detected.
NOACFT		FILE12(22). Estimated number of aircraft detected.
TIMDET		FILE12(23). Time of detection.
ISENST		FILE12(26). Sensor type.

d. Logical Flow (Figure III-3-B-4):

- (1) Block 1. Bring in contents of data file 36.
- (2) Block 2. Get from data file 16 the sequence numbers for the sources of information to be used in the development of the sensing report numbers.
- (3) Block L4. Read the first intelligence information card, which contains source and target units, time to enter Intelligence and Control Model,

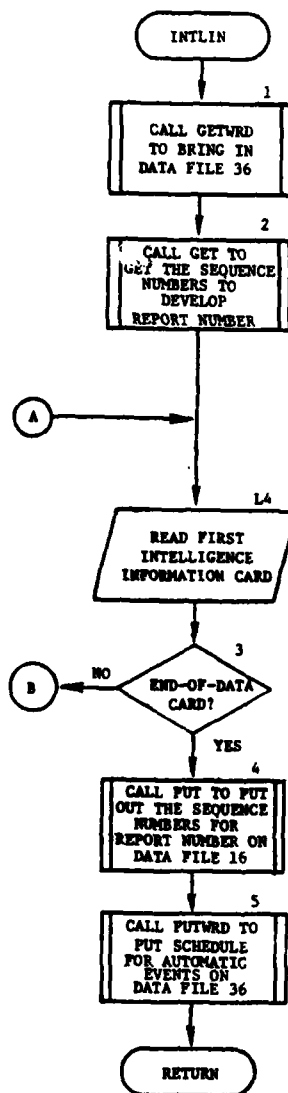


Figure III-3-B-4. Routine INTLIN (Continued on Next Page)



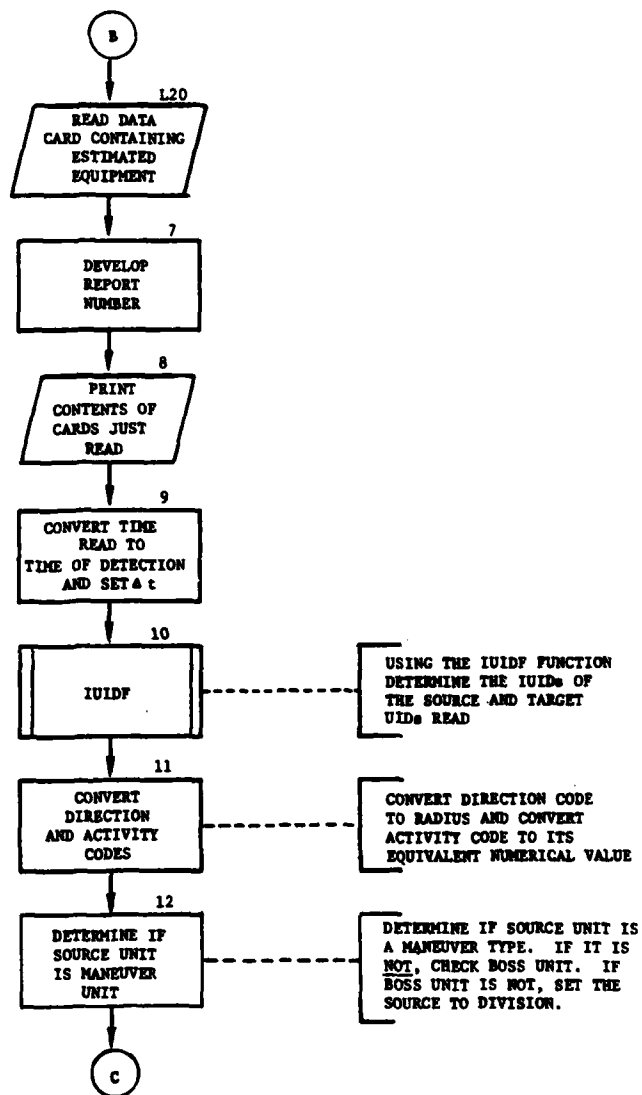


Figure III-3-B-4. Routine INTLIN (Continued)

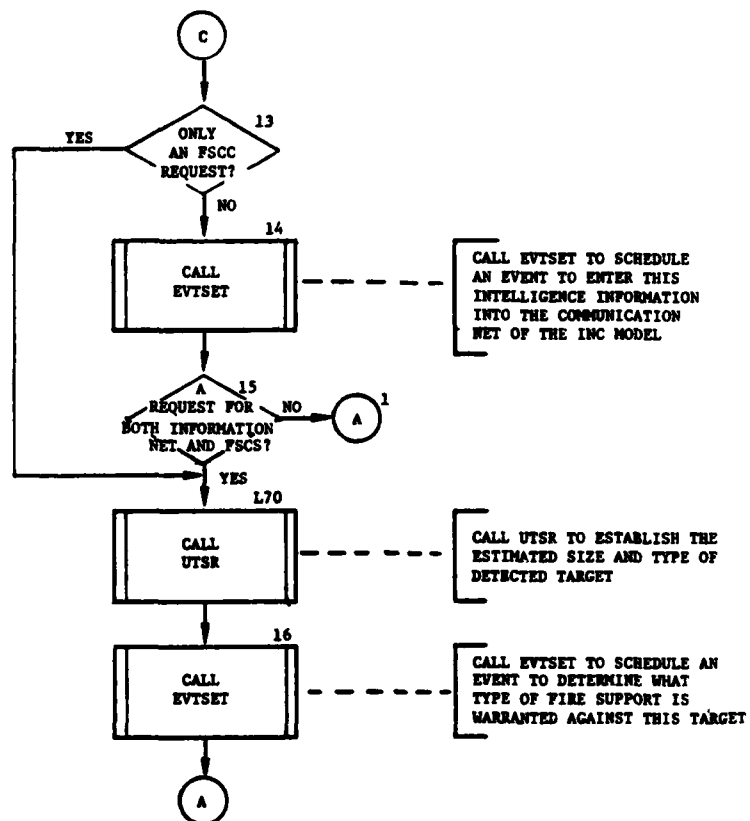


Figure III-3-B-4. Routine INTLIN (Concluded)

and time of detection, source type, estimated location, estimated activity, and if activity is estimated to be movement, the estimated direction and rate of movement.

(4) Block 3. If this is not an end-of-data card (i.e., a blank card), control goes to block L20.

(5) Block 4. Since all data have been read, save source sequence numbers by putting them on data file 16.

(6) Block 5. Since some Intelligence and Control events have been scheduled by this routine, save the schedule for automatic events by putting it on data file 36. Return control to the calling routine.

(7) Block L20. Read the second card containing the estimated equipment items contained in the target unit.

(8) Block 7. Develop the report number for the sensing report, which will also be used on the printout.

(9) Block 8. Print the contents of the two intelligence information cards just read.

(10) Block 9. Convert the time read into time of detection (i.e., time in minutes from beginning of game). This time is also used as the time increment to schedule the entry of this information into the Intelligence and Control Model.

(11) Block 10. Utilizing the routine IUIDF, determine the identification numbers of the source and target units read on the first card of intelligence information.

(12) Block 11. Convert the direction code read into the proper radian equivalent. Convert the activity code read into its proper numerical equivalent. (See DATA statement in listing).

(13) Block 12. Determine if the specified source unit is a maneuver unit by checking its unit type designator. If it is not a maneuver unit, check the unit type designator of its superior unit. If neither unit is a maneuver unit, then set the specified source unit's division as the indicated source unit on the sensing report.

(14) Block 13. Check the code indicating where this intelligence information is to enter the Intelligence and Control Model; if it is a request for only fire support coordination processing, control goes to block L70.

(15) Block 14. Call EVTSET to schedule an event to enter this intelligence information into the communication network of the Intelligence and Control Model.

(16) Block 15. If this is not a request to enter this information into both the information communication network and fire support coordination center, control goes to block L4.

(17) Block L70. Call UTSR to establish the estimated size and type of the target unit entered in the data cards.

(18) Block 16. Call EVTSET to schedule an event to determine what type of fire support should be requested against this target; transfer control to block L4.

10. ROUTINES IUIDF, UTSR, EVTSET. These routines are described in Section IV.

## APPENDIX C

### OUTPUT DESCRIPTIONS FOR OPERATING INSTRUCTIONS

The operating instructions (OPRINS) process initially lists all unit identification codes in the Blue and Red forces. Figure III-3-C-1 depicts the format of the output produced by the operating instructions process. In the figure, an alphabetical character (descriptor) designates an appropriate group of lines explained below.

<u>Output Descriptor</u>	<u>Explanation</u>
------------------------------	--------------------

A

This line reflects the unit identification code of the last unit in the Blue force (288), and the first unit in the Red force (655).

B

INTLAN DATA CARDS: There are two lines reflecting intelligence information loaded in routine INTLAN of the operating instructions. The report number (4800011) reflects the source of information (48) and a unique sequential report number (11). This unique report number is generated by adding 10 to a source type code, multiplying the result by 100,000 and adding a sequential report number. For a description of source type codes, see Appendix A to this chapter.

BOTH: Both indicates that this report is entered both in the intelligence processing and communication network and in the fire support allocation logic. Other possible entries here are INTL to indicate that this report is entered in the intelligence processing and communication network only, or FSCC to indicate that this report is entered in the fire support allocation logic only.

B82MEZBD and RZ130ZTK are unit identification codes of the first node of the communication network and the target unit respectively.

The next two entries (1 500) are the day and hour/minute in which the information is to be entered into the Intelligence and Control Model. The next two entries are the estimated X and Y coordinates of the target unit.

The next entry (DEFN) is the estimated activity of the target unit. See Appendix A of this chapter for all estimated activity mnemonics and their interpretations.









Output  
Descriptor

Explanation

The last two entries of this line reflect the estimated rate of movement (in meters per minute) and the estimated direction of movement of the target unit.

The second line of this report reflects the estimated number of personnel, vehicles, tanks, armored personnel carriers, artillery tubes, artillery missiles, air defense guns, air defense missiles, and aircraft of the target unit.

C

EVTSET, event report: The first line of this report gives the time that the event occurs within this period and the event number. The next four lines are an image of a 35-word record of data file 12. The first-line entries (per column) are:

IUID of the division to which this data is sent  
The intelligence operation code  
Not used  
Not used  
Index number  
IUID of target number  
X coordinate of unit  
Y coordinate of unit  
Estimated type of target unit  
Estimated size of target unit

The second line gives:

Estimated activity of target unit  
Estimated movement rate of target unit (meters per minute)  
Estimated movement direction of target unit (tenths of radians)  
Estimated personnel strength of target unit  
Estimated number of vehicles in target unit  
Estimated number of tanks in target unit  
Estimated number of armored personnel carriers in target unit  
Estimated number of artillery tubes in target unit  
Estimated number of artillery missiles in target unit  
Estimated number of air defense guns in target unit

The 10 words on the third line are:

Estimated number of air defense missiles in target unit

Output  
Descriptor

Explanation

Estimated number of aircraft in target unit  
Time of detection of target unit (since start of  
game  
Detector code

The remaining nine words are not used.

- D This entry indicates that routine UTSR was called to estimate the size of the target unit.
- E Sensor data cards. A description of these cards appears in Appendix A to this chapter.
- F Unattended ground sensor (UGS) field composition and location cards. See Appendix A to this chapter for a description of these cards.
- G UGS data report. The first three lines of this report reflect UGS coverage for personnel, wheeled, and tracked vehicles. The first entry gives the sensitive radius of the sensor for the respective target; the second entry gives the cumulative sum of the square of the sensitivity radius of all sensors in the UGS field. The next entry gives the area of the UGS field, and the last entry is the number of sensors of the type reflected by this line. The last line of this report is the same as the third line with the exception of SUM. On this line, SUM reflects the ratio of the sums of the area of coverage of all UGS sensors in the UGS field to the area of the UGS field.
- H Moving target indicator (MTI) radar report. This printout is a report of data file 20 records of MTI radar sensor data. Each line has the following entries:

Record number  
Sensor type code  
Key to sensor type parameter table.  
IUID of sensor's first node unit  
Last assigned report number  
Sensor status flag  
Sensor X coordinate  
Sensor Y coordinate

Output  
Descriptor

Explanation

Sensor Z coordinate.  
Minimum range.  
Maximum range.  
Horizontal orientation (radius from X axis).  
Horizontal search sector (radians).  
Minimum radial target velocity.  
Azimuth autoscan rate (milliradians per centiminutes).  
Range autoscan depth (meters).

The remaining fields are not used.

I

Counterbattery radar report. This printout is a report of data file 20 records of individual countermortar/counterbattery radar data. Each line has the following entries:

Record number.  
Sensor type code.  
Key to sensor type parameter table.  
IUID of sensors first node unit.  
Last assigned report number.  
Sensor status flag.  
Sensor X coordinate.  
Sensor Y coordinate.  
Horizontal center azimuth (radians from X axis).  
Horizontal search sector (radians).  
Site mask angle (radians).  
Vertical beam separation (radians).

The remaining fields are not used.

J

Unattended ground sensor (UGS) report. This printout is a report of data file 20 records of UGS data. Each line has the following entries:

Record number.  
Sensor type code.  
Key to sensor type parameter table.  
IUID of sensor first node unit.  
Last assigned report number.  
Sensor status flag.  
X coordinate of first UGS field corner.  
Y coordinate of first UGS field corner.  
X coordinate of second UGS field corner.  
Y coordinate of second UGS field corner.  
X coordinate of third UGS field corner.  
Y coordinate of third UGS field corner.  
X coordinate of fourth UGS field corner.  
Y coordinate of fourth UGS field corner.

The remaining fields are not used.

Output  
Descriptor

Explanation

- |   |  |
|---|--|
| K | PLOAD, penetration data report. This report reflects the maximum range in meters beyond the FEBA to which missions may be automatically requested within the model for Blue attack helicopter missions, Blue artillery fire missions, Red attack helicopter missions, and Red artillery fire missions. |
| L | Blue and Red sorties report. Line one reflects the maximum number of all weather sorties for each 6-hour period starting at midnight for the Blue force. Line two reflects that data for the Red force.  |

APPENDIX D

SOURCE LISTINGS FOR ORDERS INPUT PROCESSOR OPERATING INSTRUCTIONS

(AVAILABLE UNDER SEPARATE COVER)

III-3-D-1

